This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problems Mailbox.

L Number	Hits	Search Text	DB	Time stamp
1	1701	345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743	USPAT	2003/02/03 10:09
2	175	hierarchical ADJ menu	USPAT	2003/02/03 10:09
3	772	setting ADJ screen	USPAT	2003/02/03 10:09
4	1924	rotation ADJ operation	USPAT	2003/02/03 10:10
5	630	push ADJ operation	USPAT	2003/02/03 10:11
6	3788	slide ADJ bar	USPAT	2003/02/03 10:12
7	166	multiple ADJ markers	USPAT	2003/02/03 10:12
8	27	(345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu)	USPAT	2003/02/03 10:13
9	0	(setting ADJ screen) and ((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu))	USPAT	2003/02/03 10:13
10	13	((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu)) and setting	USPAT	2003/02/03 10:13
11	21	((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu)) and selecting	USPAT	2003/02/03 10:14
12	2	(((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu)) and selecting) and adjusting	USPAT	2003/02/03 10:14
13	2	(((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu)) and selecting) and rotation	USPAT	2003/02/03 10:14
14	2	(((345/810 or 345/812.or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu)) and selecting) and push	USPAT	2003/02/03 10:15
15	0	(((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu)) and selecting) and slider	USPAT	2003/02/03 10:16
16	0	(slide ADJ bar) and ((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743) and (hierarchical ADJ menu))	USPAT	2003/02/03 10:17

17	67	(slide ADJ bar) and menus	USPAT	2003/02/03 10:17
18	48	((slide ADJ bar) and menus) and moves	USPAT	2003/02/03 10:18
19	30	(((slide ADJ bar) and menus) and moves) and stop	USPAT	2003/02/03 10:18
20	10	((((slide ADJ bar) and menus) and moves) and stop) and marker	USPAT	2003/02/03 10:18

L Number	Hits	Search Text	DB	Time stamp
1	6312	345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156	USPAT	2003/02/03 14:26
2	1924	rotation ADJ operation	USPAT	2003/02/03 14:27
3	13	(345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (rotation ADJ operation)	USPAT	2003/02/03 14:31
4	3788	slide ADJ bar	USPAT	2003/02/03 14:31
5	4	(rotation ADJ operation) and (slide ADJ bar)	USPAT	2003/02/03 14:32
6	4721	345/156	USPAT	2003/02/03 14:33
7	9	(rotation ADJ operation) and 345/156	USPAT	2003/02/03 14:33
8	3	((rotation ADJ operation) and 345/156) and slide	USPAT	2003/02/03 14:33
9	0	(((rotation ADJ operation) and 345/156) and slide) and stop	USPAT	2003/02/03 14:33
10	0	(((rotation ADJ operation) and 345/156) and slide) and puch	USPAT	2003/02/03 14:33
11	2	(((rotation ADJ operation) and 345/156) and slide) and push	USPAT	2003/02/03 14:44
12	14245	stop ADJ position	USPAT	2003/02/03 14:45
13	119	(slide ADJ bar) and (stop ADJ position)	USPAT	2003/02/03 14:45
14	630	push ADJ operation	USPAT	2003/02/03 14:45
15	0	((slide ADJ bar) and (stop ADJ position)) and (push ADJ operation)	USPAT	2003/02/03 14:46
16	0	(rotation ADJ operation) and ((slide ADJ bar) and (stop ADJ position))	USPAT	2003/02/03 14:46
17	221	move ADJ marker	USPAT	2003/02/03 14:47
18	176	setting ADJ item	USPAT	2003/02/03 14:47
19	0	(move ADJ marker) and (setting ADJ item)	USPAT	2003/02/03 14:48
20	6	(345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (move ADJ marker)	USPAT	2003/02/03 14:48
21	6	(345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (setting ADJ item)	USPAT	2003/02/03 14:48
22	0	(rotation ADJ operation) and ((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (move ADJ marker))	USPAT	2003/02/03 14:48
23	0	(rotation ADJ operation) and ((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (setting ADJ item))	USPAT	2003/02/03 14:49
24	279	(stop ADJ position) and marker	USPAT	2003/02/03 14:49

25	630	push ADJ operation	USPAT	2003/02/03 14:50
26	0	((stop ADJ position) and marker) and (push ADJ operation)	USPAT	2003/02/03 14:53
27	0	((rotation ADJ operation) and ((slide ADJ bar) and (stop ADJ position))) and (push ADJ operation)	USPAT	2003/02/03 14:53

L Number	Hits	Search Text	DB	Time stamp
1	6312	345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156	USPAT	2003/02/03 14:26
2	1924	rotation ADJ operation	USPAT	2003/02/03 14:27
3	13	(345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (rotation ADJ operation)	USPAT	2003/02/03 14:31
4	3788	slide ADJ bar	USPAT	2003/02/03 14:31
5	4	(rotation ADJ operation) and (slide ADJ bar)	USPAT	2003/02/03 14:32
6	4721	345/156	USPAT	2003/02/03 14:33
7	9	(rotation ADJ operation) and 345/156	USPAT	2003/02/03 14:33
8	3	((rotation ADJ operation) and 345/156) and slide	USPAT	2003/02/03 14:33
9	0	(((rotation ADJ operation) and 345/156) and slide) and stop	USPAT	2003/02/03 14:33
10	0	(((rotation ADJ operation) and 345/156) and slide) and puch	USPAT	2003/02/03 14:33
11	2	(((rotation ADJ operation) and 345/156) and slide) and push	USPAT	2003/02/03 14:44
12	14245	stop ADJ position	USPAT	2003/02/03 14:45
13	119	(slide ADJ bar) and (stop ADJ position)	USPAT	2003/02/03 14:45
14	630	push ADJ operation	USPAT	2003/02/03 14:45
15	0	((slide ADJ bar) and (stop ADJ position)) and (push ADJ operation)	USPAT	2003/02/03 14:46
16	0	(rotation ADJ operation) and ((slide ADJ bar) and (stop ADJ position))	USPAT	2003/02/03 14:46
17	221	move ADJ marker	USPAT	2003/02/03 14:47
18	176	setting ADJ item	USPAT	2003/02/03 14:47
19	0	(move ADJ marker) and (setting ADJ item)	USPAT	2003/02/03 14:48
20	6	(345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (move ADJ marker)	USPAT	2003/02/03 14:48
21	6	(345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (setting ADJ item)	USPAT	2003/02/03 14:48
22	0	(rotation ADJ operation) and ((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (move ADJ marker))	USPAT	2003/02/03 14:48
23	0	(rotation ADJ operation) and ((345/810 or 345/812 or 345/817 or 345/818 or 345/819 or 345/820 or 345/830 or 345/833 or 345/841 or 345/764 or 345/853 or 345/828 or 345/829 or 345/743 or 345/156) and (setting ADJ item))	USPAT	2003/02/03 14:49
24	279	(stop ADJ position) and marker	USPAT	2003/02/03 14:49

25	630	push ADJ operation	USPAT	2003/02/03 14:50
26	0	((stop ADJ position) and marker) and (push ADJ operation)	USPAT	2003/02/03 14:50

US005856827A

United States Patent [19]

Sudo

[11] Patent Number: 5,856,827

[45] Date of Patent: Jan. 5, 1999

[54] **PORTABLE TELEVISION WITH RECOGNIZABLY DIFFERENT LOW/HIGH HIERARCHY DISPLAYS**

[75] Inventor: **Fukuharu Sudo**, Tokyo, Japan

[73] Assignee: **Sony Corporation**, Tokyo, Japan

[21] Appl. No.: **805,364**

[22] Filed: **Feb. 24, 1997**

[30] **Foreign Application Priority Data**

Feb. 26, 1996 [JP] Japan 8-065377

[51] **Int. Cl.**[6] **G06F 15/00**

[52] **U.S. Cl.** **345/352**; 345/157; 345/184; 345/348; 345/352

[58] **Field of Search** 345/348, 352, 345/353, 157, 184

[56] **References Cited**

U.S. PATENT DOCUMENTS

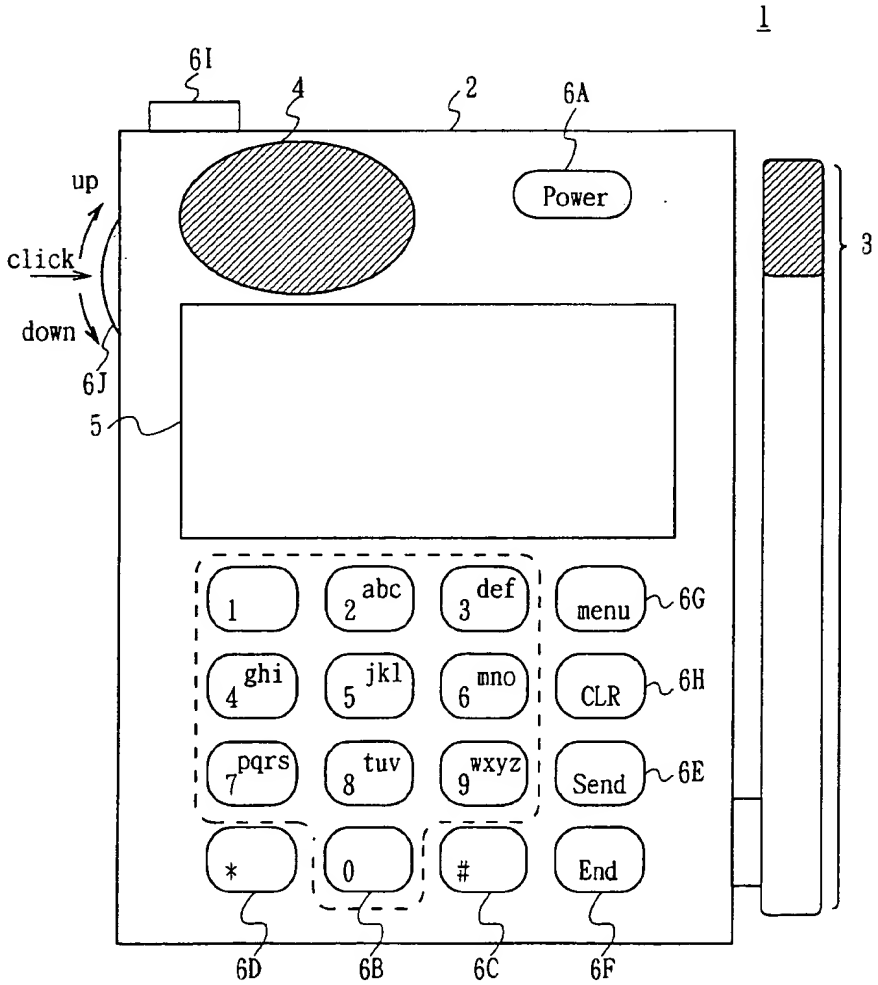
5,491,784 2/1996 Dougles et al. 345/352
5,692,145 11/1997 Nakanishi 345/348

Primary Examiner—Huynh Ba
Attorney, Agent, or Firm—Jay H. Maioli

[57] **ABSTRACT**

A portable communication apparatus provides a display on which the user is able to easily find out the object item at the time of displaying a list of the plural items. In the case where the functional items are separated into some groups and formed into hierarchical structure, the display layout is changed for each hierarchy at the time of displaying the list of the items. Therefore, by viewing the differences among the display layouts, the user can know which hierarchy is currently being displayed and can easily find out the object item.

11 Claims, 17 Drawing Sheets



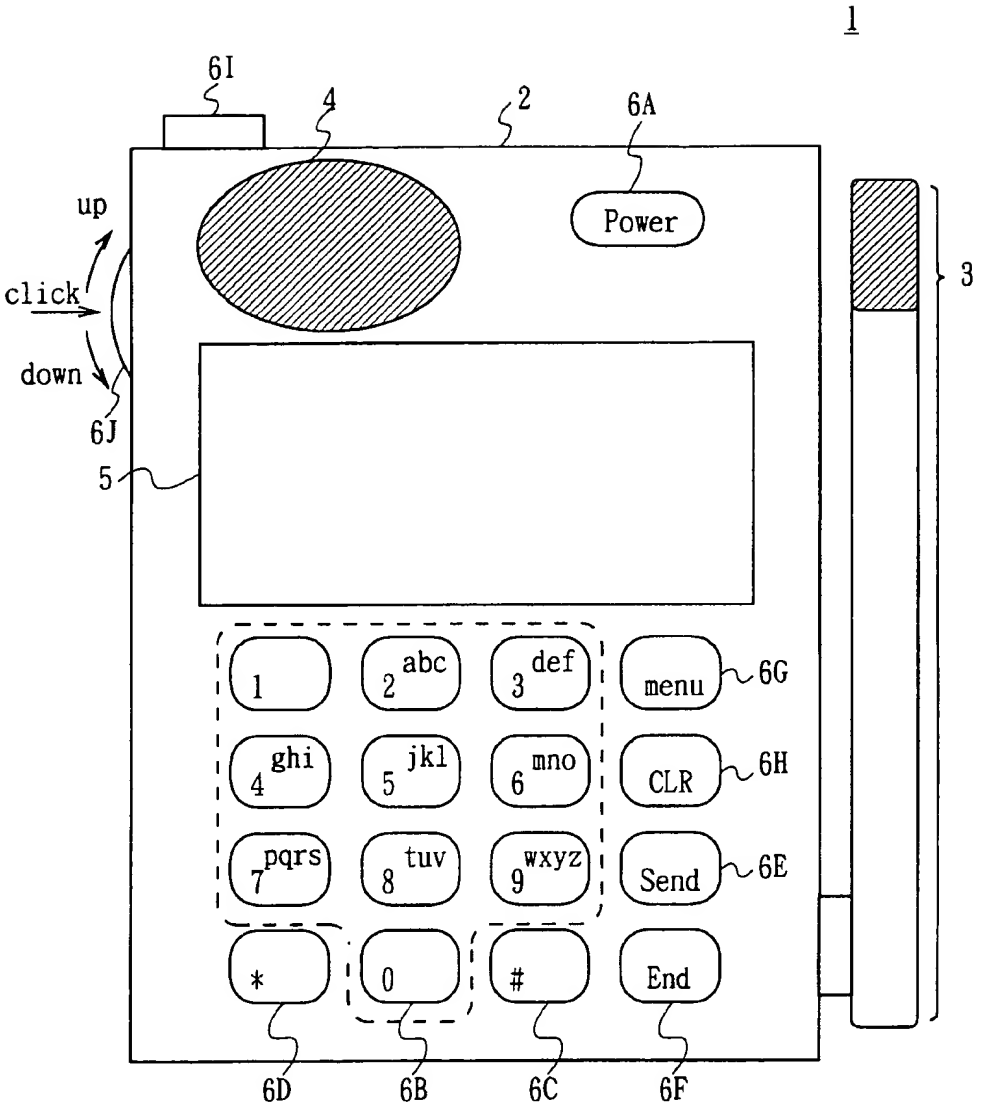

1
6I
4
2
6A
Power
up
click
down
6J
5
3
1
2 abc
3 def
menu
6G
4 ghi
5 jkl
6 mno
CLR
6H
7 pqrs
8 tuv
9 wxyz
Send
6E
*
0
#
End
6D
6B
6C
6F

FIG. 1

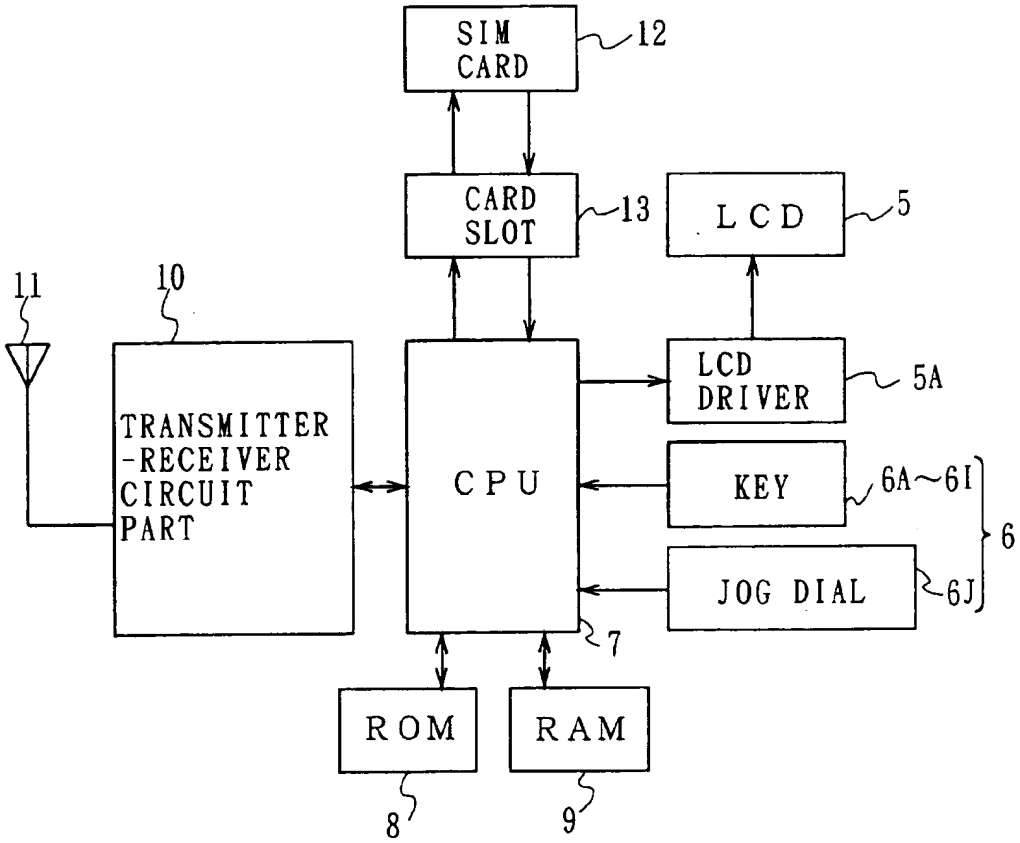

SIM CARD
12
CARD SLOT
13
LCD
5
11
10
TRANSMITTER -RECEIVER CIRCUIT PART
CPU
LCD DRIVER
5A
KEY
6A~6I
6
JOG DIAL
6J
7
ROM
RAM
8
9

FIG. 2

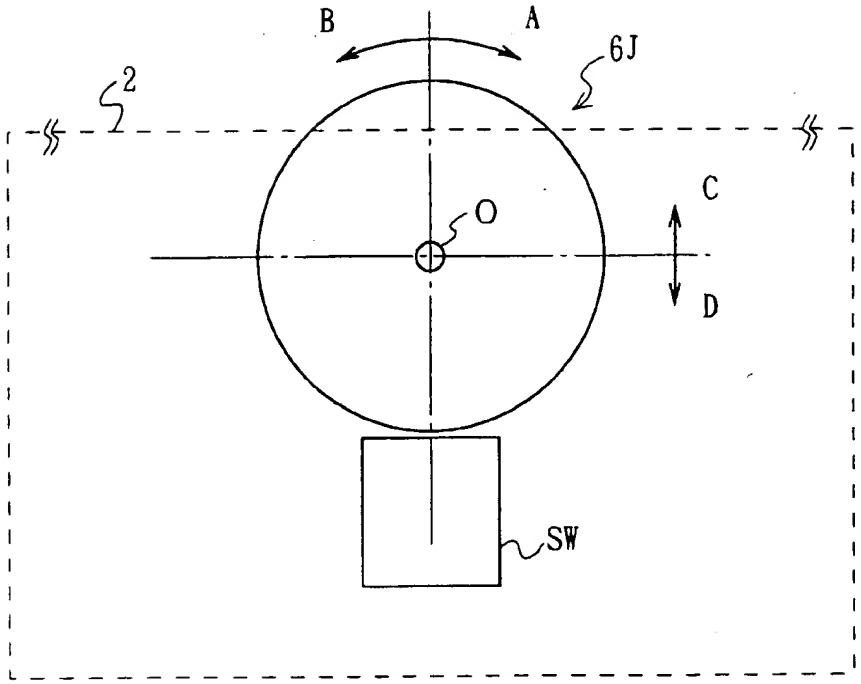


FIG. 3

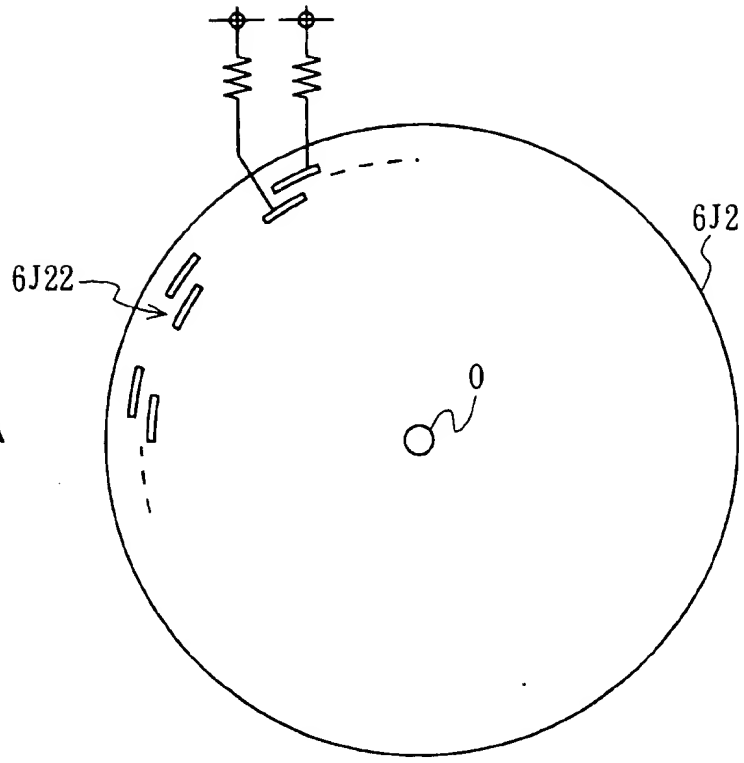


FIG. 4A

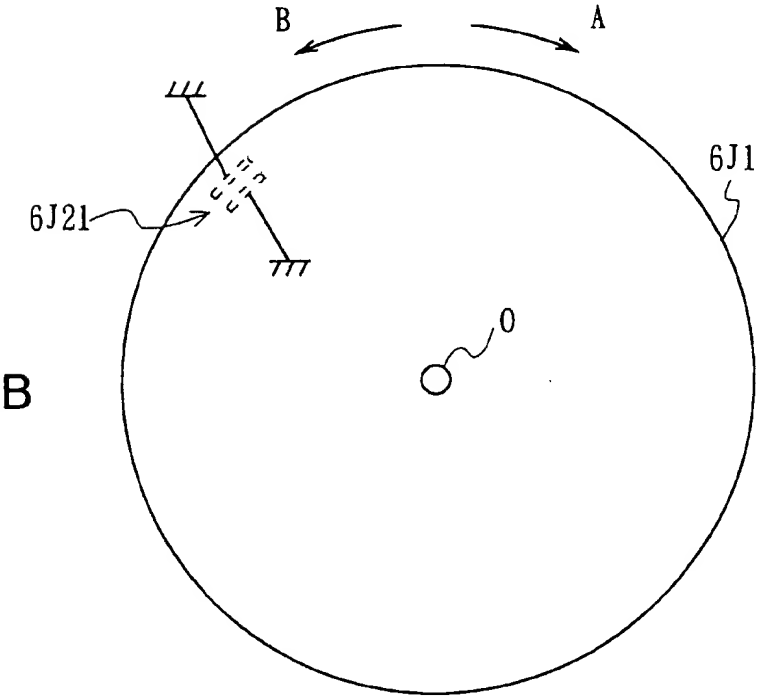


FIG. 4B

FIG. 5A

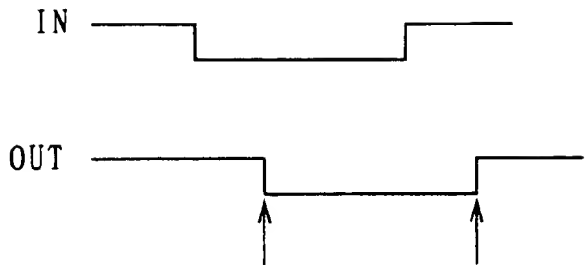
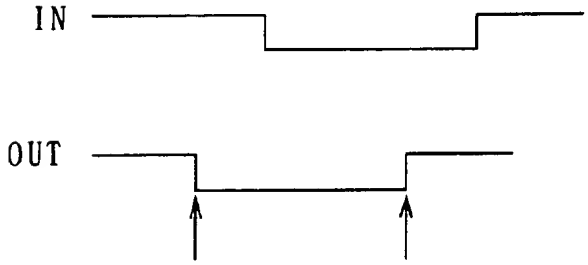


FIG. 5B



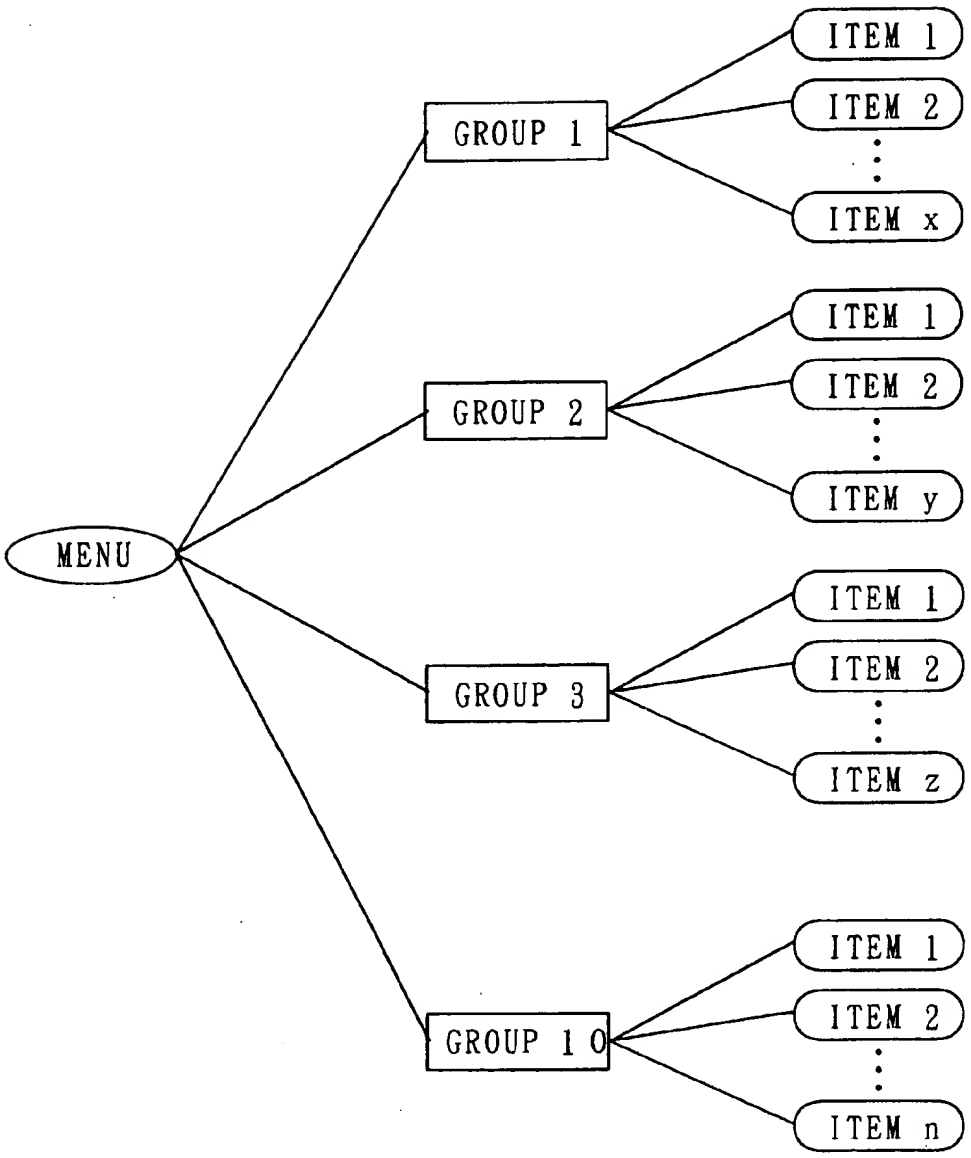
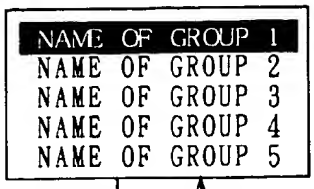

MENU
GROUP 1
GROUP 2
GROUP 3
GROUP 10
ITEM 1
ITEM 2
ITEM x
ITEM 1
ITEM 2
ITEM y
ITEM 1
ITEM 2
ITEM z
ITEM 1
ITEM 2
ITEM n

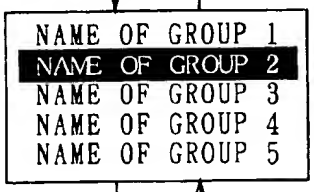
FIG. 6

FIG. 7A



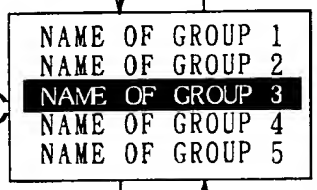
DOWN UP

FIG. 7B



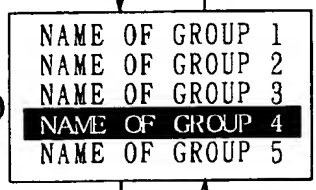
DOWN UP

FIG. 7C



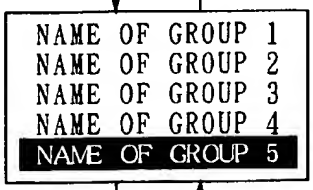
DOWN UP

FIG. 7D



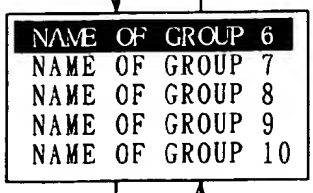
DOWN UP

FIG. 7E



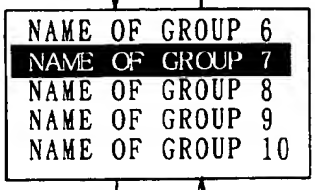
DOWN UP

FIG. 7F



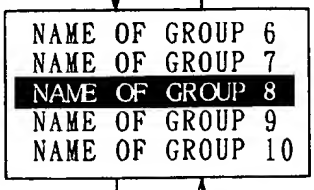
DOWN UP

FIG. 7G



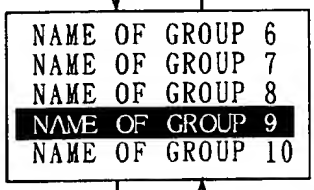
DOWN UP

FIG. 7H



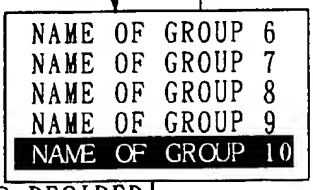
DOWN UP

FIG. 7I



DOWN UP

FIG. 7J



GROUP IS DECIDED BY CLICKING

FIG. 7K

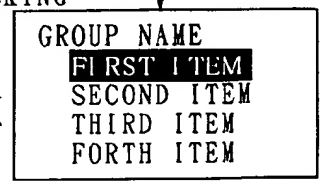
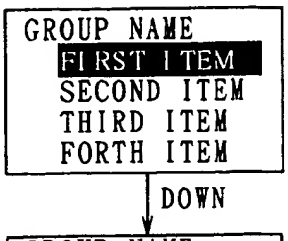
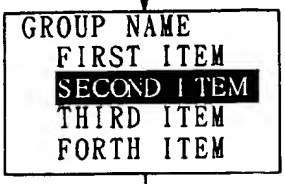


FIG. 8A



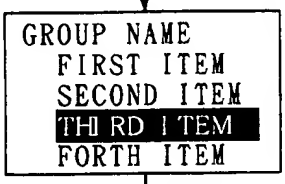
DOWN

FIG. 8B



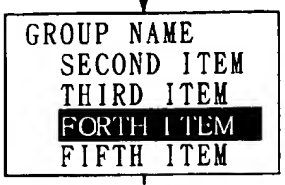
DOWN

FIG. 8C



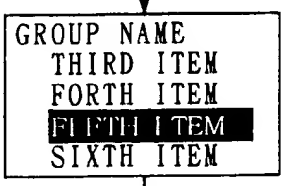
DOWN

FIG. 8D



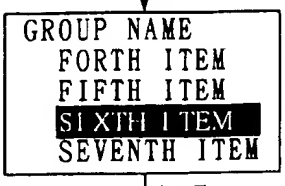
DOWN

FIG. 8E



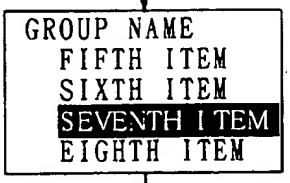
DOWN

FIG. 8F



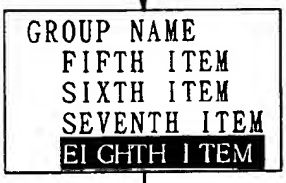
DOW

FIG. 8G



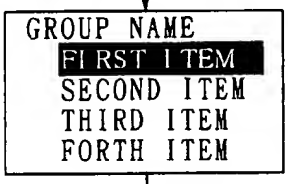
DOWN

FIG. 8H



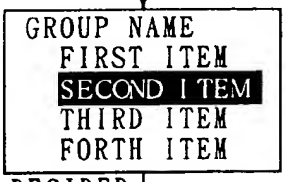
DOWN

FIG. 8I



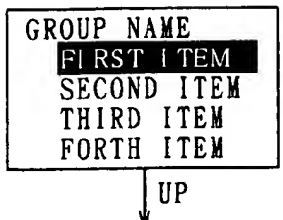
DOWN

FIG. 8J



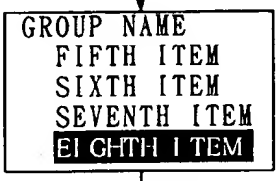
ITEM IS DECIDED BY CLICKING

FIG. 9A



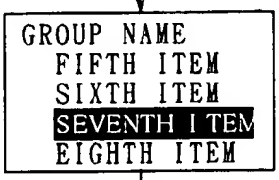
UP

FIG. 9B



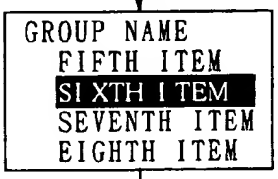
UP

FIG. 9C



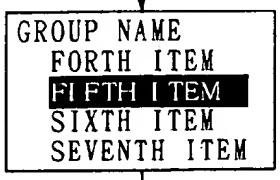
UP

FIG. 9D



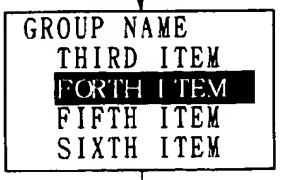
UP

FIG. 9E



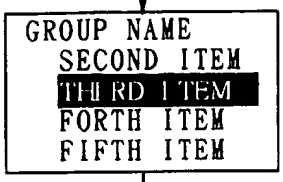
UP

FIG. 9F



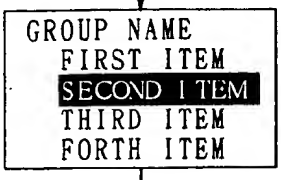
UP

FIG. 9G



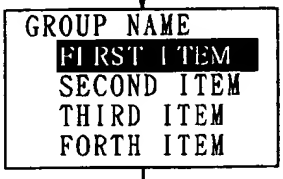
UP

FIG. 9H



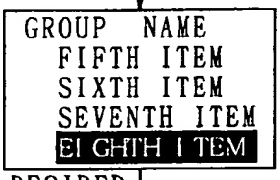
UP

FIG. 9I



UP

FIG. 9J



ITEM IS DECIDED BY CLICKING

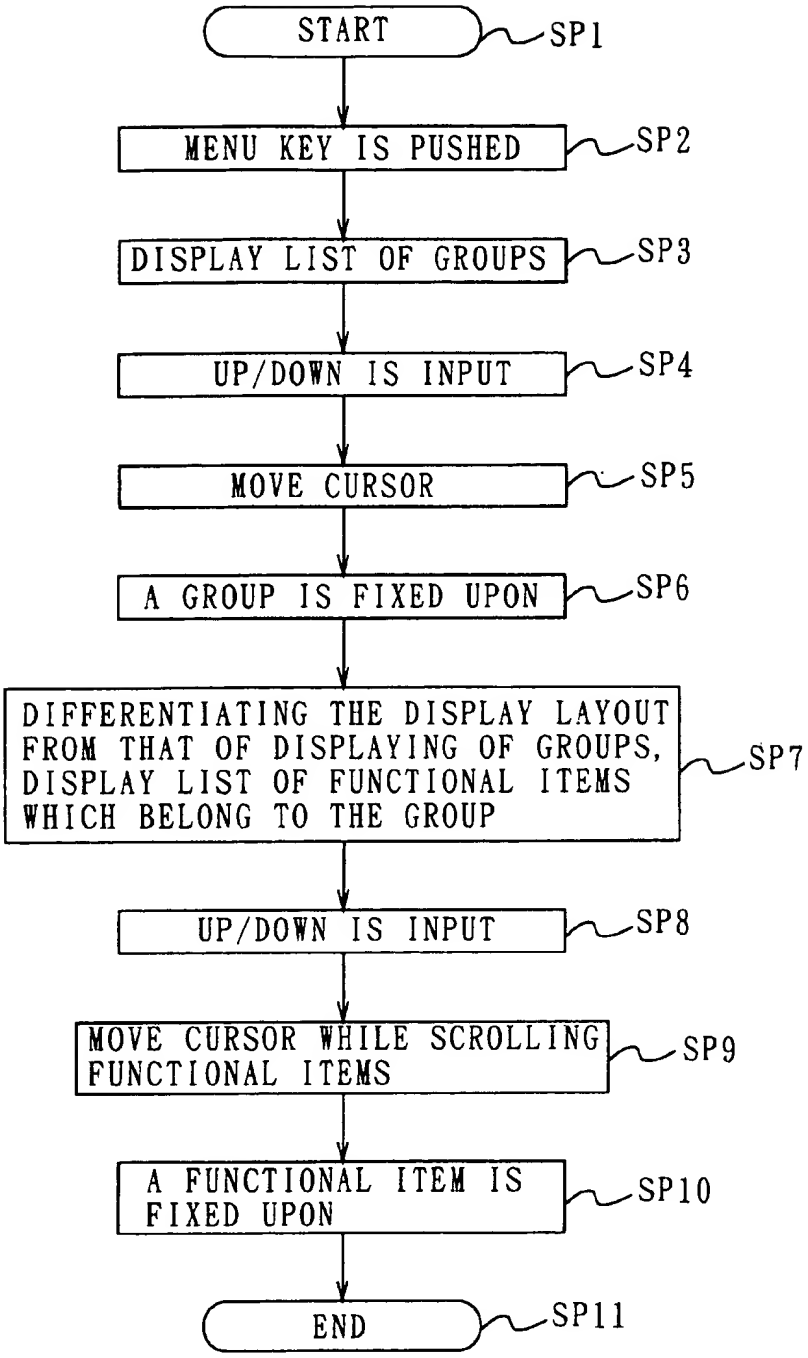

START
SP1
MENU KEY IS PUSHED
SP2
DISPLAY LIST OF GROUPS
SP3
UP/DOWN IS INPUT
SP4
MOVE CURSOR
SP5
A GROUP IS FIXED UPON
SP6
DIFFERENTIATING THE DISPLAY LAYOUT FROM THAT OF DISPLAYING OF GROUPS, DISPLAY LIST OF FUNCTIONAL ITEMS WHICH BELONG TO THE GROUP
SP7
UP/DOWN IS INPUT
SP8
MOVE CURSOR WHILE SCROLLING FUNCTIONAL ITEMS
SP9
A FUNCTIONAL ITEM IS FIXED UPON
SP10
END
SP11

FIG. 10

GROUP 1	ITEM 1 OF GROUP 1	SIX ITEMS BELONGING TO GROUP 1
	ITEM 2 OF GROUP 1	
	ITEM 3 OF GROUP 1	
	ITEM 4 OF GROUP 1	
	ITEM 5 OF GROUP 1	
	ITEM 6 OF GROUP 1	
GROUP 2	ITEM 1 OF GROUP 2	SEVEN ITEMS BELONGING TO GROUP 2
	ITEM 2 OF GROUP 2	
	ITEM 3 OF GROUP 2	
	ITEM 4 OF GROUP 2	
	ITEM 5 OF GROUP 2	
	ITEM 6 OF GROUP 2	
	ITEM 7 OF GROUP 2	
GROUP 3	ITEM 1 OF GROUP 3	THREE ITEMS BELONGING TO GROUP 3
	ITEM 2 OF GROUP 3	
	ITEM 3 OF GROUP 3	

FIG. 11

NAME OF GROUP 3	ITEM 1 OF GROUP 3
	ITEM 2 OF GROUP 3
	ITEM 3 OF GROUP 3

FIG. 18

FIG. 12A

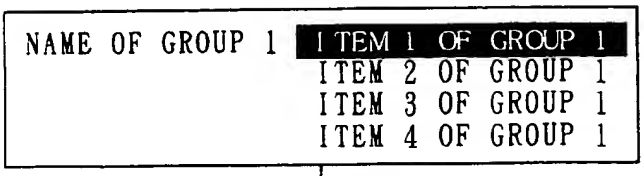


FIG. 12B

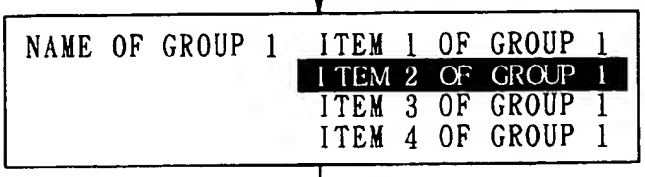


FIG. 12C

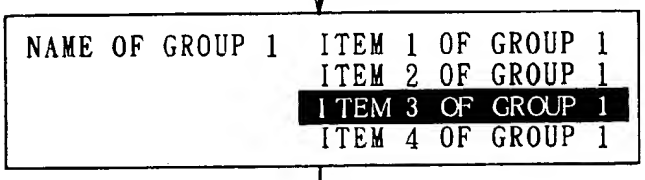


FIG. 12D

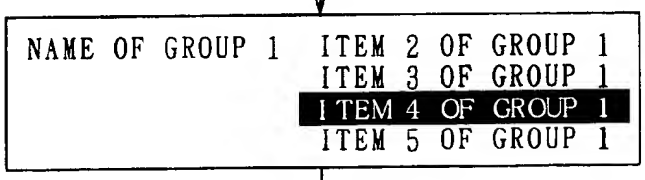


FIG. 12E

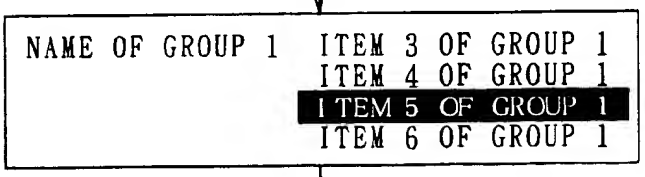


FIG. 12F

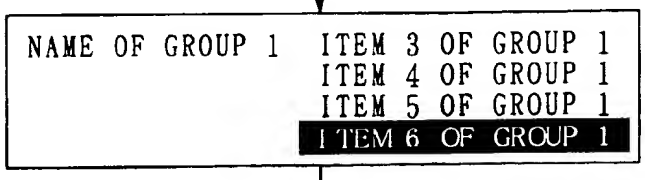


FIG. 12G

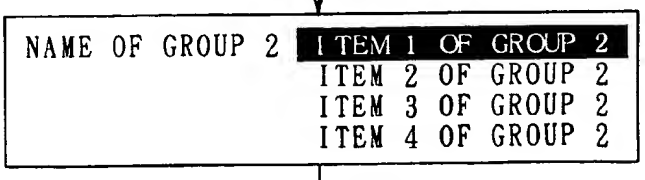


FIG. 13H

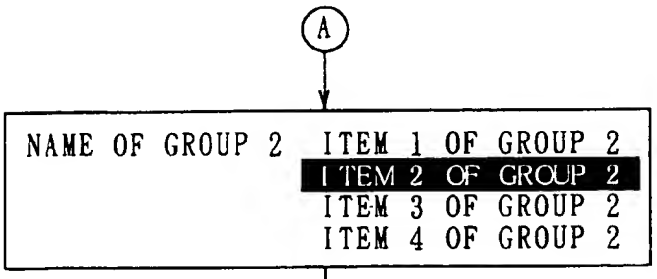
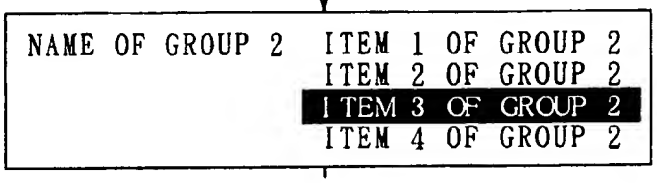
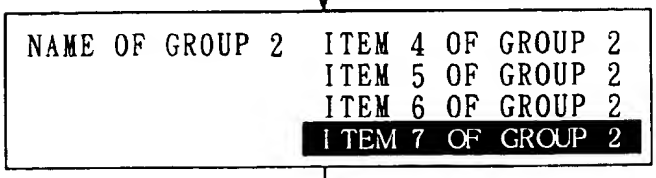


FIG. 13I



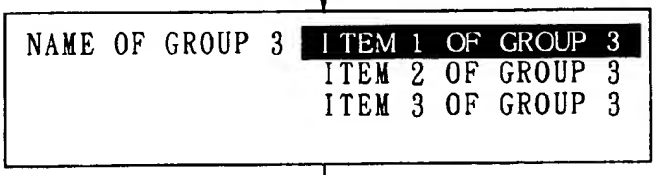
DOWN, DOWN, . . .

FIG. 13J



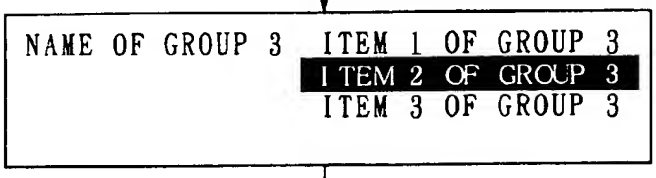
DOWN

FIG. 13K



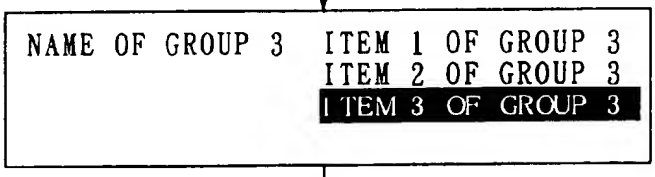
DOWN

FIG. 13L



DOWN

FIG. 13M



DOWN

FIG. 13N

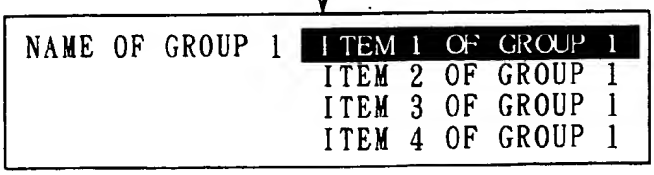
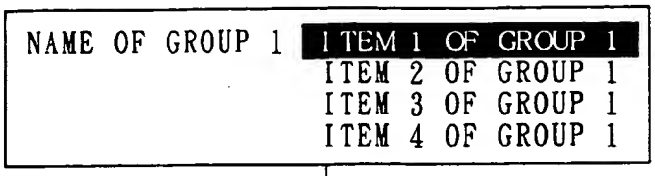
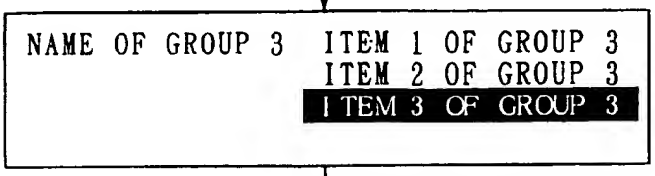


FIG. 14A



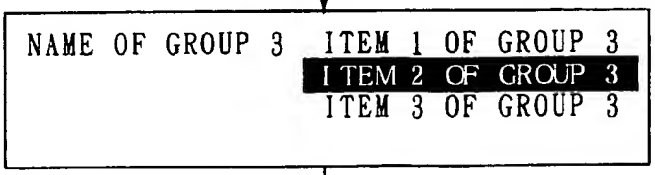
UP

FIG. 14B



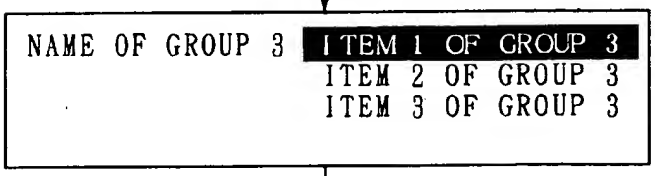
UP

FIG. 14C



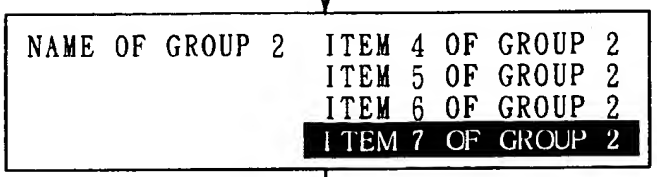
UP

FIG. 14D



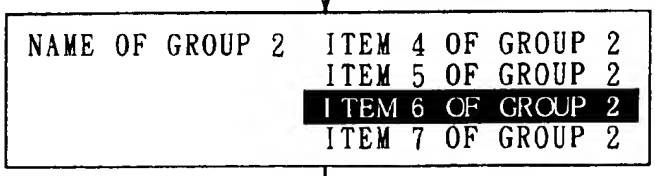
UP

FIG. 14E



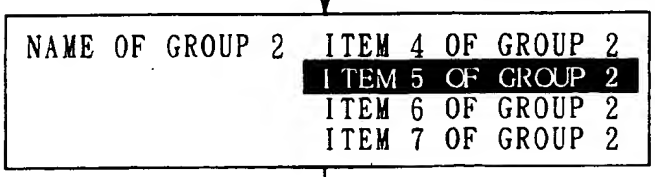
UP

FIG. 14F



UP

FIG. 14G



UP

FIG. 15H

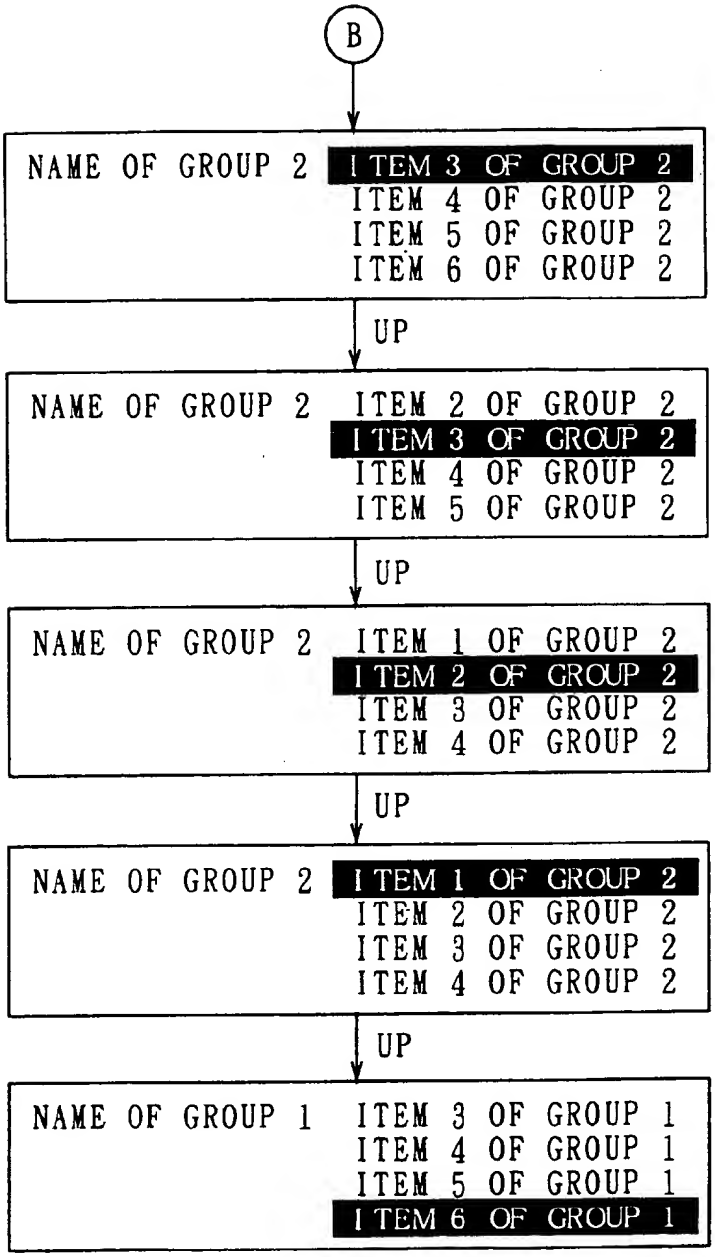


FIG. 15I

FIG. 15J

FIG. 15K

FIG. 15L

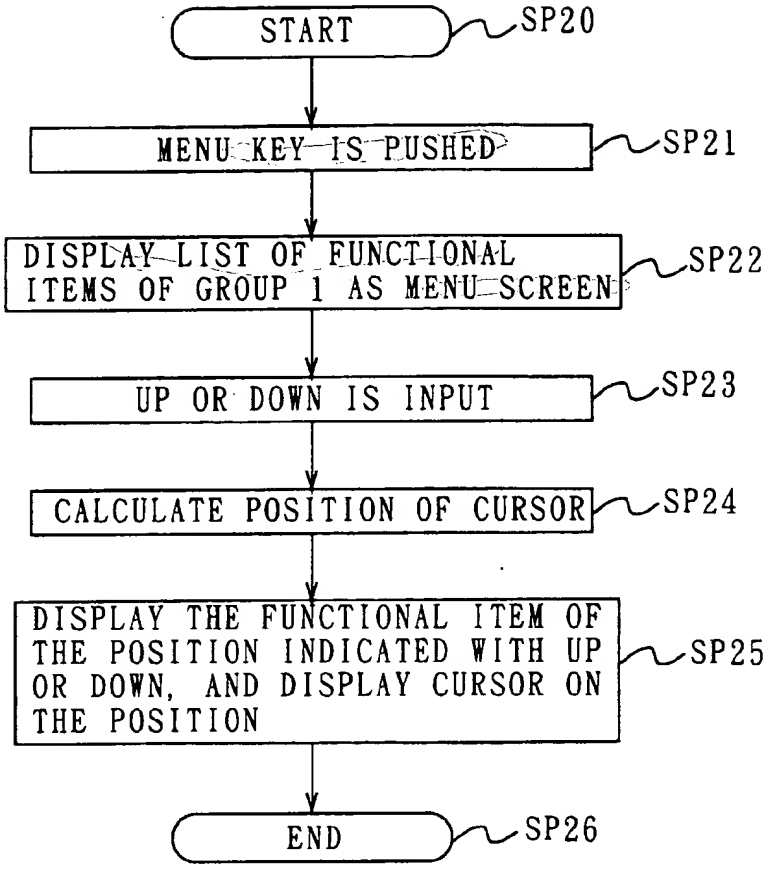

START
SP20
MENU KEY IS PUSHED
SP21
DISPLAY LIST OF FUNCTIONAL ITEMS OF GROUP 1 AS MENU SCREEN
SP22
UP OR DOWN IS INPUT
SP23
CALCULATE POSITION OF CURSOR
SP24
DISPLAY THE FUNCTIONAL ITEM OF THE POSITION INDICATED WITH UP OR DOWN, AND DISPLAY CURSOR ON THE POSITION
SP25
END
SP26

FIG. 16

FIG. 17A

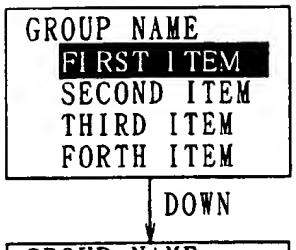
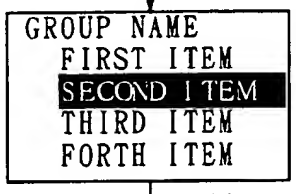


FIG. 17B



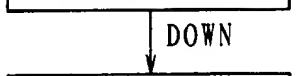
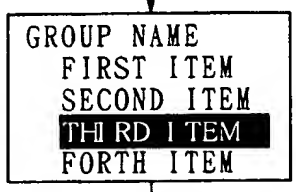


FIG. 17C



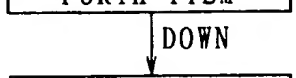
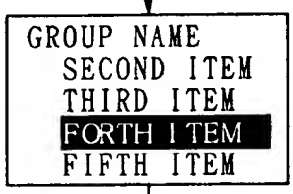


FIG. 17D



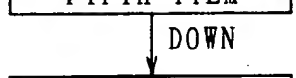
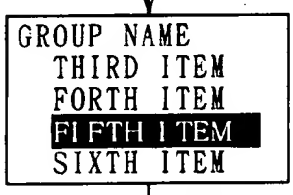


FIG. 17E



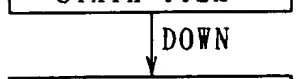
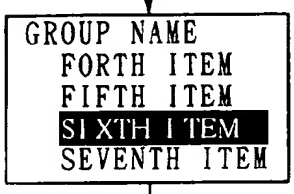


FIG. 17F



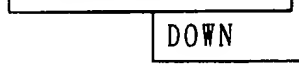
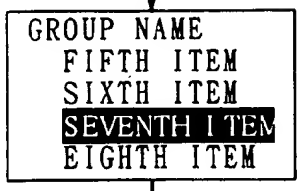


FIG. 17G



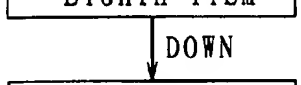
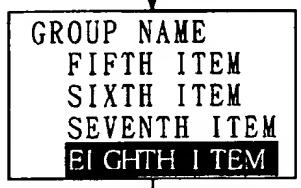


FIG. 17H



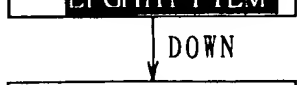
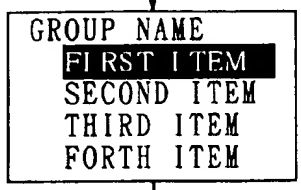


FIG. 17I



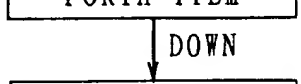
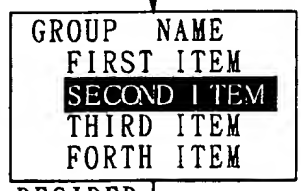


FIG. 17J



PORTABLE TELEVISION WITH RECOGNIZABLY DIFFERENT LOW/HIGH HIERARCHY DISPLAYS

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates to a portable communication apparatus, and more particularly, is suitably applied to a portable telephone apparatus which displays a list of plural items registered in a menu.

2. Description of the Related Art

In recent years, the diffusion of portable telephone apparatuses is remarkable, and, in proportion to it, a great variety of functions are added to the portable telephone apparatuses. There are various kinds of such functions, as follows. Telephone directory function for registering the telephone number for each name, short-message transmission function for transmitting a short telegraphic message, transfer function for transferring a call-in toward another terminal, and call out/in limit function for limiting call-out and call-in, etc.

Usually, these functions are registered in a menu. The menu is read and a desired item is selected in turn, so that an arbitrary setting can be obtained in accordance with a user's desire.

By the way, since the functional items to be displayed have increased and the number of the rows and the number of the columns of the liquid crystal display which has been provided as a display means are not sufficient, the items are separated into groups which are respectively comprised of the related items, and the menu which is provided in the portable telephone apparatus is hierarchically constructed, in general. So, when the stated functional item is to be set, the group to which the functional item belongs is first selected, and then the list of the items which belong to the group is displayed. Thereafter, the desired functional item is found during scrolling the list of the items, and the cursor is adjusted to that portion to select the functional item, and the setting is performed.

However, in the conventional menu displaying, ways of displaying are the same with respect to a display of a group of high hierarchy and a display of functional items of low hierarchy. For this reason, there is such inconvenience that the hierarchy becomes unclear to the user and so the desired functional item can not be found, at the time of searching of the desired functional item. This problem occurs more often as the number of hierarchies is increased and the structure is complicated, and it is considered that the problem occurs still more often when the number of the functional items is hereafter increased in accordance with increasing the functions.

SUMMARY OF THE INVENTION

In view of the foregoing, an object of this invention is to provide a portable communication apparatus which provides a display on which a user can easily find an object item, at the time of displaying a list of plural items.

The foregoing object and other objects of the invention have been achieved by the provision of a portable communication apparatus in which, in the case where the plural items have been separated into groups and formed into hierarchical structure, a display layout is changed for each hierarchy at the time of displaying the list of the items. Thereby, the user can know which hierarchy's items are currently being displayed by viewing the difference among the display layouts, and therefore can easily find the desired object item.

Besides, in the present invention, in the case of displaying the list of the plural items, the plural items are separated into groups, and the group name to which the displayed items belong is displayed on the display means. In this way, the user become capable of easily finding out the object item, taking the group name as a guide.

The nature, principle and utility of the invention will become more apparent from the following detailed description when read in conjunction with the accompanying drawings in which like parts are designated by like reference numerals or characters.

BRIEF DESCRIPTION OF THE DRAWINGS

In the accompanying drawings:

FIG. **1** is a schematic diagram showing the general constitution of a portable telephone apparatus according to one embodiment of the present invention;

FIG. **2** is a block diagram showing a constitution of the circuit which is provided within the portable telephone apparatus;

FIG. **3** is a schematic diagram showing a mechanism of the jog dial schematically;

FIGS. **4**A and **4**B are schematic diagrams showing a constitution of a rotary encoder;

FIGS. **5**A and **5**B are diagrams showing the output waveform of the rotary encoder;

FIG. **6** is a schematic diagram explaining the menu structure according to the first embodiment;

FIGS. **7**A to **7**K, **8**A to **8**J, and **9**A to **9**J are schematic diagrams explaining the menu displaying according to the first embodiment;

FIG. **10** is a flow chart showing the procedure of the menu displaying according to the first embodiment;

FIG. **11** is a schematic diagram explaining the menu structure according to the second embodiment;

FIGS. **12**A to **12**G, **13**H to **13**N, **14**A to **14**G, and **15**H to **15**L are schematic diagrams explaining the menu displaying according to the second embodiment;

FIG. **16** is a flow chart showing the procedure of the menu displaying according to the second embodiment;

FIGS. **17**A to **17**J are schematic diagrams explaining the case where the display layout has been changed by changing the character font; and

FIG. **18** is a schematic diagram explaining the case where the broken line for indicating the boundary of the group has been displayed.

DETAILED DESCRIPTION OF THE EMBODIMENT

Preferred embodiments of this invention will be described with reference to the accompanying drawings:

At first, the overall configuration and the circuit configuration of a portable telephone apparatus **1** will be explained with reference to FIGS. **1** and **2**. The portable telephone apparatus **1** is roughly composed of a main body **2** of the apparatus and an arm microphone **3** which is attached to the side of the main body so as to be freely rotated (opened and closed). Therefore, while the apparatus is not in use, the size of the portable telephone apparatus **1** can be further reduced by closing the arm microphone **3**.

The arm microphone **3** has a function for negating the operation of keys other than the specific key in a closed state (hereinafter, referred to as a key-lock function), in addition

to a function for controlling on-hook or off-hook in response to an opening or closing operation. In this connection, the key-lock condition can be canceled by opening the arm microphone **3**.

Such key-lock function prevents operation keys from being inadvertently pressed to start malfunction while the portable telephone apparatus **1** is put in a pocket or a bag.

On the other hand, the main body **2** of the apparatus is provided with the various operation keys and a signal processing circuit. A speaker **4** paired with the arm microphone **3**, a liquid crystal display (LCD) **5**, and plural operation keys **6** to which various functions are assigned are provided on the surface of the main body **2** of the apparatus. A central processing unit (CPU) **7** which uses those as input/output devices is built into the main body.

The CPU **7** controls the liquid crystal display **5** via a liquid crystal display driver **5A** to display information according to an instruction input from the operation keys **6** with a character font of an appropriate size. In addition, the CPU **7** controls a transmitter-receiver circuit part **10** to transmit and receive information to and from a base station apparatus via an antenna **11** connected to the part **10**. The CPU **7** operates based on programs stored in a read only memory (ROM) **8** and data read into a random access memory (RAM) **9**.

In this connection, a card socket **13** is connected to the CPU **7**, and management information on subscribers is read from a subscriber ID card **12** (in this embodiment, a subscriber identity module (SIM) card) inserted into the card socket **13** and utilized for controlling.

In this connection, the liquid crystal display **5** comprises a matrix of pixels having, for instance, 40 dots×97 dots, and these dots can be used to display information with two types of fonts. One of the fonts is a small font for displaying one character with, for instance, vertical 7 dots×lateral 5 dots, while the other is a large font for displaying one character with, for instance, vertical 15 dots×lateral 8 dots. Therefore, the use of the small font enables vertical five characters and lateral sixteen characters (that is, characters of 5 rows×16 columns) to be displayed, while the use of the large font enables vertical two characters and lateral ten characters (that is, characters of 2 rows×10 columns) to be displayed.

As a general rule, the large font is used to display the characters input by the user, while the small font is used to display the message from the apparatus. However, if the number of characters input by the user exceeds a predetermined number (for instance, about twenty characters) , the font size is automatically switched from the large font to the small font.

Such a font-switching function enables large characters to be input with the contents of input confirmed while the number of the input characters is small, thereby incorrect inputs can be reduced. Moreover, if the number of the characters is large, the input information on the same item can be checked on a screen, and the contents of input can be easily understood.

Next, the operation keys **6** provided on the main body **2** of the apparatus will he explained. In the case of this embodiment, the operation keys **6** comprises ten keys: a power key **6A**, numerical keys **6B** of "0" to "9", a "#" key **6C**, a "*" key **6D**, a send key **6E**, an end key **6F**, a menu key **6G**, a clear key **6H**, a record key **6I**, and a jog dial **6J**. The operation keys **6A** to **6H** are disposed on the front surface of the main body **2** of the apparatus, and the remaining two operation keys **6I** and **6J** are disposed on the sides of the main body **2** of the apparatus.

The main functions assigned to respective operation keys are as follows. At first, the power key **6A** is a key for supplying power to an internal circuit in the main body **2** of the apparatus. The power is turned on by the first push-down operation, and the power is turned off by the second push-down operation. However, in the case where the personal identity number (PIN) has not been input by the user during thirty seconds after turning-on of the power through the Power key **6A**, the CPU **7** detects this to automatically turn the power off. This prevents the power from remaining turned on due to malfunction.

Next, ten numerical keys **6B** will be explained. These ten numerical keys **6B** are used to input not only numerals but also alphabet. In the case of this embodiment, the plural alphabetical characters are assigned to each of the eight numerical keys "2" to "9" excluding "0" and "1", thereby alphabet can be input through these keys. For instance, "a" to "c" are assigned to the "2" key, "d" to "f" are assigned to the "3" key, and the other alphabetical characters are likewise assigned to the other keys.

At the time of inputting alphabet, the first character can be input by pushing the same key once, the second character can be input by pushing the same key twice, and the third character can be input by pushing the same key three times, in sequence.

The send key **6E** is a key for inputting the operation start command at the time of calling a telephone number, manually input through the numerical keys **6B**, or a destination telephone number, selected from the screen of telephone-directory list. In addition, the send key **6E** is used to call the history of past dials.

The end key **6F** is a key for inputting a speech end command. In this connection, the speech end command can also be input by closing the arm microphone **3**.

The menu key **6G** is used to switch the screen, displayed on the liquid crystal display **5**, between the initial screen and the menu screen. Usually, the portable telephone apparatus **1** displays time, and so on, as the initial screen. When the menu key **6G** is pushed in this state, the display can be switched to the menu screen for setting the various functional items. Besides, if the menu key **6G** is pushed while the menu screen is being displayed, the display can immediately return to the initial screen regardless of whatever screen is being displayed.

Besides, the menu key **6G** is assigned a function as the only operation key which can cancel the key-lock state. That is, as described above, the apparatus is normally in the key-lock state while the arm microphone **3** is closed, however, when the menu key **6G** is pushed, it is able to escape from key-lock state and to move to key-active state.

The record key **6I** is a key which is used to record conversations and to reproduce the recorded conversations, and the key **6I** is mounted on the top surface of the main body **2** of the apparatus opposite to the arm microphone **3** so as to be operated by the user's hand holding the main body **2** of the apparatus.

Finally, the jog dial **6J** having the central function within the ten operation keys will be explained. The jog dial **6J** is provided in the upper part of the side of the main body **2** of the apparatus opposite to the arm microphone **3** and near the speaker **4**, and can be operated by the user's hand holding the main body **2** of the apparatus just as the record key **6I**.

The jog dial **6J** can be independently operated in the circumferential direction and the radial direction. As shown in FIG. **3**, it comprises a disc-like member (consisting of a rotary encoder) which rotates around a rotation axis O in the

circumferential direction (as shown by the arrows A and B), a slide plate (not shown) which can slide in the radial direction (as shown by the arrows C and D), and a slide switch SW.

In this connection, the slide plate and the slide switch SW are urged in the direction shown by the arrow C. Besides, the rotation axis O is fixed to the slide plate. When the jog dial **6J** is pushed in the direction shown by the arrow D, the rotary encoder can slide integrally with the slide plate to push down the slide switch SW in order to turn the switch on. The CPU **7** discriminates whether or not the jog dial **6J** has been clicked, by detecting the ON/OFF state of the slide switch SW.

The rotary encoder which slides integrally with the slide plate comprises two discs **6J1** and **6J2**, as shown in FIGS. **4A** and **4B**. The disc **6J1** is a movable member laminated on the top surface of the disc **6J2**, and is mounted so as to rotate relative to the disc **6J2** fixed to the slide plate. The movable disc **6J1** has a pair of opposite electrodes **6J21**. When assembled, the opposite electrodes **6J21** slidably contact twenty pairs of opposite electrodes **6J22** provided along the circumference of the disc **6J2**. In this connection, the opposite electrodes **6J22** provided on the fixed disc **6J2**, which are disposed on the inner and outer circumferential sides, respectively, are slightly offset from each other.

Therefore, as to the potential which is outputted from the opposite electrodes **6J22**, when the jog dial **6J** is rotated in the direction shown by the arrow A, the potential of the inner side falls to the earth potential first, as shown in FIG. **5A**. On the contrary, when the jog dial **6J** is rotated in the direction shown by the arrow B, the potential of the outer side falls to the earth potential first, as shown in FIG. **5B**. using this nature, by detecting which of the inner side potential and the outer side potential falls first, the rotational direction of the jog dial **6J** is detected. Besides, the rotation quantity of the jog dial **6J** can be detected by counting the number of pulses output from the outer side electrodes.

Next, a typical example of operation using the jog dial **6J** will be explained. By performing up or down operation of the jog dial **6J** to the circumferential direction while various list screens are being displayed on the liquid crystal display **5**, the cursor which is being displayed on the liquid crystal display **5** can be moved to the vertical direction. In this case, by pushing the jog dial **6J** to the radial direction in this state (hereinafter, this is referred to as clicking), the CPU **7** can be instructed to read detailed information on the item on which the cursor is located or the item of the low hierarchy of the clicked item.

Besides, by rotating the jog dial **6J** in the circumferential direction during a conversation, the loudness of the receiving sound can be adjusted. Also, by clicking the jog dial **6J** during a conversation, the muting can be performed.

In view of the stream of realizing the high-level function of the portable telephone apparatus of recent years, the portable telephone apparatus **1** has various functions, such as the telephone directory function, the transfer function, the short-message transmission function, and the call out/in limit function. These functions are registered in the menu, and by calling out the menu screen, setting can be performed at any time according to the user's needs.

The menu in the portable telephone apparatus **1** will be now explained. However, only the structure of the menu and its displaying method will be explained here, and the explanation of the contents of the functional items will be omitted. As shown in FIG. **6**, the menu of the portable telephone apparatus **1** is first separated into ten groups of the related items, and then the respective functional items are contained under the respective groups; in this way, the menu is formed on the basis of so-called hierarchical structure. For instance, the functional items "1 to x" are contained in the low hierarchy of the group 1, and the functional items "1 to y" are contained in the low hierarchy of the group 2.

By the way, in the case of the menu which has hierarchical structure in this way, usually, there is such a possibility that the user can not recognize which hierarchy is being displayed when the list is displayed, as described in the above section "Description of the Related Art". For this reason, in the portable telephone apparatus **1**, the screen layout is changed in accordance with the depth of hierarchy, so that it is facilitated to recognize which hierarchy in the menu the user in at the present time.

The concrete displaying method will be now explained with reference to the layout of the menu screen shown in FIGS. **7A** to **7K**, **8A** to **BJ**, and **9A** to **9J**.

In the portable telephone apparatus **1**, when the menu key **6G** is pushed, the respective group names of the groups 1 to 5 are displayed on the liquid crystal display **5** as the first page of the menu screen, as shown in FIG. **7A**. At this time, the cursor for indicating the portion which is being selected at present (that is, the portion of white-black inversion displaying) is positioned on the first line of the group 1, and each group name is displayed in a left-aligned manner with respect to the top of each line.

By rotating the jog dial **6J** upwardly or downwardly in the first-page displaying (hereinafter, this is referred to as up or down operation), the cursor can be moved upwardly or downwardly. For instance, when down operation of the jog dial **6J** is once performed in the state shown in FIG. **7A**, the cursor can be lowered by one line and moved to the place of the group 2, as shown in FIG. **7B**. When down operation of the jog dial **6J** is further performed once in this state, the cursor can be further lowered by one line and moved to the place of the group 3, as shown in FIG. **7C**. When down operation of the jog dial **6J** is sequentially performed in like manner, the cursor can be moved to the forth line of the group 4 or the fifth line of the group 5, as shown in FIGS. **7D** and **7E**.

When up operation of the jog dial **6J** is sequentially performed, the cursor can be sequentially moved upwardly line by line, as shown in FIGS. **7E** to **7A**.

As shown in FIG. **7E**, when down operation of the jog dial **6J** is once performed in such a state that the cursor lies in the place of the group 5, which is the end of the first page of the menu screen, the groups 6 to 10 are displayed as the second page of the menu screen this time, as shown in FIG. **7F**. At this time, as with the first page, each group name is displayed in a left-aligned manner with respect to the top of each line, in the second page too.

Besides, in the second page too, by performing up or down operation of the jog dial **6J**, the cursor can be moved upwardly or downwardly. For instance, when down operation of the jog dial **6J** is once performed in the state shown in FIG. **7F**, the cursor can be moved to the second line of the group 7, as shown in FIG. **7G**. When down operation of the jog dial **6J** is further performed once in this state, the cursor can be moved to the third line of the group 8, as shown in FIG. **7H**. When down operation of the jog dial **6J** is sequentially performed in like manner, the cursor can be moved to the forth line of the group 9 or the fifth line of the group 10, as shown in FIG. **7I** and FIG. **7J**.

When up operation of the jog dial **6J** is sequentially performed in contrast with this, the cursor can be sequentially moved upwardly line by line, as shown in FIGS. **7J** to **7F**.

In the portable telephone apparatus **1**, when the cursor is adjusted to the place of the desired group name and the jog dial **6J** is clicked, the items of the low hierarchy of that group are displayed. For instance, when the jog dial **6J** is clicked in such a state that the cursor lies in the place of the group **10** as shown in FIG. **7J**, the functional items which belong to the group **10** are displayed as shown in FIG. **7K**.

At this time, in the first line of the liquid crystal display **5**, the name of the high hierarchy to which those functional items belong is displayed, that is, the group name is displayed in this case, while in the second line to the fifth line, the respective functional items are displayed. By virtue of displaying the name of the high hierarchy in the fifth line of the liquid crystal display **5** in this manner, the user is able to look at this and to easily understand whether he is in the low hierarchy or not at present, and he is also able to easily understand which hierarchy he is under.

Besides, each functional item is displayed in such a manner that it is indented by two characters from the head of each line. The display formats are changed between high hierarchy and low hierarchy in this way, therefore, the user is able to discover at a glance that he is in the low hierarchy, without reading of the group name.

Then, scroll displaying at the time of selection of the functional item in the low hierarchy is explained. When the jog dial **6J** has been clicked and functional items have been displayed for the first time, four items, the first item to the fourth item, are displayed as the functional items, and the cursor is displayed at the place of the first item, as shown in FIG. **8A**. When down operation of the jog dial **6J** is once performed in this state, the cursor can be lowered by one line and moved to the place of the second item, as shown in FIG. **8B**. When down operation of the jog dial **6J** is further performed once in this state, the cursor can be further lowered by one line and moved to the place of the third item as shown in FIG. **8C**.

By the way, as to display of functional items, page-scrolling system which has been used at the time of group displaying is not utilized, and, when down operation of the jog dial **6J** is continuously performed, the list of the functional items is continuously shifted upwardly from midway, so that the entire list of the functional items is scrolled. That is, when down operation of the jog dial **6J** is performed once in such a state that the cursor lies in the place of the third item as shown in FIG. **8C**, the cursor is not lowered in the screen (that is, the cursor is left on the fourth line from the top in the screen) and, on the contrary, the list of the functional items is shifted upwardly by one row, as shown in FIG. **8D**, so that the cursor is moved to the place of the fourth item and indicates the item. In this connection, in this case, the group name which is being displayed on the first line in the screen is not scrolled, and only the list of the functional items is scrolled.

When down operation of the jog dial **6J** is further performed in the state shown in FIG. **8D**, the cursor itself is not lowered and the list of the functional items is shifted upwardly by one row in like manner, as shown in FIG. **8E**, so that the cursor is moved to the place of the fifth item and indicates the item. When down operation of the jog dial **6J** is likewise performed continuously from this time, the list of the functional items is continuously shifted upwardly row by row, and the cursor is hereby moved to the places of the sixth item and the seventh item in sequence, as shown in FIGS. **8F** to **8G**.

When down operation of the jog dial **6J** is performed in the case where the last functional item, the eighth item, is being displayed in the fifth line of the screen as shown in FIG. **8G**, the list of the functional items is not shifted this time and the cursor is lowered by one line, so that the cursor is hereby moved to the place of the eighth item, as shown in FIG. **8H**. The reason why the position of the cursor is thus changed with respect to the last functional item is to enable to easily find that it is the last functional item, on the basis of the cursor position in the screen.

When down operation of the jog dial **6J** is further performed in such a state that the cursor lies in the place of the last functional item as shown in FIG. **8H**, the first state is brought back and the first item to the fourth item are displayed, as shown in FIG. **8I**. At this time, the cursor is displayed in the second line, that is, the place of the first item in the screen. From this time, when down operation is performed in like manner, the cursor is moved in the order of FIG. **8J**, FIG. **8C**, ...

When the jog dial **6J** is clicked after the cursor has been thus adjusted to the desired functional item, entering to the setting mode of the functional item is achieved, and setting of the function can be performed. For instance, the jog dial **6J** is clicked when the cursor is placed at the second item as shown in FIG. **8J**, entering to the setting mode of the second item is achieved, and the setting can be performed.

On the contrary, in the case where up operation of the jog dial **6J** has been performed, the display of the functional items is scrolled as shown in FIG. **9**. That is, when up operation of the jog dial **6J** is performed in such a state that the functional items have been first displayed as shown in FIG. **9A**, the latter four items out of the functional items (that is, the fifth item to the eighth item) are displayed as shown in FIG. **9B**. At this time, the cursor is displayed on the fifth line from the top in the screen, that is, on the place of the eighth item.

When up operation of the jog dial **6J** is further performed in this state, the cursor is moved upwardly line by line in sequence, and moved to the places of the seventh item, the sixth item in sequence, as shown in FIGS. **9C** and **9D**.

When up operation of the jog dial **6J** is further performed in the state shown in FIG. **9D**, the cursor is not raised in the screen (that is, it is left on the second line from the top in the screen) and, on the contrary, the list of the functional items is shifted downwardly by one row, and the cursor is hereby moved to the place of the fifth item, as shown in FIG. **9E**. In this connection, in this case, the group name which is being displayed on the first line in the screen is not scrolled, and only the list of the functional items is scrolled.

When up operation of the jog dial **6J** is likewise performed continuously in sequence from this time, the list of the functional items is continuously shifted downwardly row by row, and the cursor is hereby moved to the places of the fourth item, the third item, and the second item in sequence, as shown in FIGS. **9F** to **9H**.

When up operation of the jog dial **6J** is performed in the case where the first of the functional items is being displayed on the second line of the screen as shown in FIG. **9H**, the functional items are not shifted and the cursor is raised by one line this time, so that the cursor is hereby moved to the place of the first item, as shown in FIG. **9I**. The reason why the position of the cursor is thus changed with respect to the first functional item is to enable to easily find that it is the first of the functional items, on the basis of the cursor position in the screen.

When up operation of the jog dial **6J** is performed in such a state that the cursor lies in the place of the first functional item as shown in FIG. **9I**, the eighth item is displayed again

and the cursor is displayed whereon as shown in FIG. 9J. When up operation is likewise performed from this time, the cursor is moved in the sequence of FIG. 9C, FIG. 9D, . . .

When the jog dial 6J is clicked after the cursor has been thus adjusted to the desired functional item, entering to the setting mode of the functional item is achieved as stated above, so that setting of the function can be performed.

The display of these menu screens are performed under the display controlling of the CPU 7; the procedure of that time will be explained referring to the flow chart shown in FIG. 10.

First, at the step SP2 following the step SP1, the menu key 6G is pushed by the user. Accepting this, the CPU 7 proceeds to the following step SP3 in order to display the list of the groups which are the high hierarchy of the menu on the liquid crystal display 5. At this time, the group 1 to the group 5 are displayed as the first page of the menu screen.

After displaying, the user operates the jog dial 6J so as to input up or down, at the following step SP4. Accepting this, the CPU 7 proceeds to the following step SP5 in order to move the cursor which is being displayed in the screen. At this time, in the case where the moving quantity which is specified with the up or down operation ranges to the second page of the menu screen, the display is switched to the groups 6 to 10 which are the second page, and the cursor is moved (that is, the display of the groups is performed in such a manner that the pages are turned, and the cursor is moved).

At the step SP6, when the jog dial 6J is clicked after the cursor has been adjusted to the place of the desired group, this group is fixed upon, and then the CPU 7 proceeds to the following step SP7.

At the step SP7, the list of the functional items which lie in the low hierarchy of the fixed group is displayed, where the display layout of the display is differentiated from that of the previous display of the groups. Specifically, the group name is displayed on the first line of the screen and the functional items are displayed on the second line to the fifth line of the screen, as described above. Besides, each functional item is displayed such that it is indented from the head of each line.

After the functional items have been thus displayed, the user operates the jog dial 6J so as to input up or down, at the following step SP8. Accepting this, the CPU 7 proceeds to the following step SP9 in order to move the cursor which is being displayed in the screen. At this time, in the case where such an up-or-down command is received that the cursor is to be moved to the portion other than the functional item which is being displayed at present on the screen, the list of the functional items is sequentially shifted so as to display the specified item portion, and the cursor is moved to this item portion. That is, with respect to the functional items, the desired functional item is not displayed in the page-turning manner like the manner of group display, but is displayed by sequentially shifting the list of the functional items.

At the step SP10, when the user clicks the jog dial 6J after the cursor has been adjusted to the place of the desired functional item, this functional item is fixed upon, and the CPU 7 enters the setting mode for setting this functional item. When this is completed, the CPU 7 proceeds to the following step SP11, and the procedure of the menu display is ended.

With the above constitution, in the case of this embodiment, when the menu key 6G is pushed, the list of the groups which are the high hierarchy of the menu is first displayed. When the jog dial 6J is clicked after the user has operated the jog dial 6J and adjusted the cursor to the place of the desired group, the list of the functional items which belong to the selected group is displayed this time. When the jog dial 6J is clicked after the user has again operated the jog dial 6J in this state and adjusted the cursor to the place of the desired functional item, the CPU 7 enters the setting mode of the selected functional item in order to perform data processing corresponding to the functional item.

By the way, in the case of this embodiment, the display layout of the case where the groups which are the high hierarchy are displayed and the display layout of the case where the functional items which are the low hierarchy of the groups are displayed have been differentiated. To put it concretely, at the time of the display of the groups, the list of the group names is displayed in a left-aligned manner and in regular order from the top of the screen, while, at the time of the display of the functional items, the group name which is the high hierarchy of the functional items is displayed on the top line of the screen, and the functional items are displayed in such a manner that they are indented from the heads of the lines.

In this way, at the time of the display of the functional items which are the low hierarchy, the name of the group which is their high hierarchy is also displayed; therefore, the user is able to easily understand that the functional items of the low hierarchy are being displayed at present by looking at the group name, and also able to easily understand to which high hierarchy the functional items being displayed belong.

Besides, at the time of the display of the functional items of the low hierarchy, each functional item is displayed in such a manner that it is indented from the head of the line so as to provide a blank portion on the head of the line; therefore, the user is able to easily understand that the functional items which are the low hierarchy are being displayed at present, by glancing at the display, without reading of the group name.

Besides, in the case of this embodiment, at the time of the display of the groups which are the high hierarchy, such page scrolling is performed that the pages are turned sheet by sheet, and, at the time of the display of the functional items which are the low hierarchy, such scrolling is performed that the functional items are sequentially shifted. In this way, the scrolling method of the case where the groups which are the high hierarchy are displayed and the scrolling method of the case where the functional items which are the low hierarchy are displayed have been differentiated; therefore, by looking at this, the user is able to understand whether the high hierarchy is being displayed or the low hierarchy is being displayed.

According to the above constitution, by differentiating between the display layouts of the case where the items of the high hierarchy are displayed and the case where the items of the low hierarchy are displayed, the user is able to easily know which hierarchy is being displayed in present, and hereby able to easily find the object item. As a result, such a portable telephone apparatus can be realized that is able to provide a display on which the object item can be easily found by the user, at the time of the display of the list of the plural items.

In the explanation of the aforementioned first embodiment, such a case has been described that the menu has been formed into hierarchical structure and hierarchically displayed in accordance with the structure. However, in this second embodiment, the menu is formed into planar structure on the basis of simple grouping, and a group's

name and items' names which belong to the group are concurrently displayed, so as to ease finding of the object item.

Referring to FIG. **11**, the menu of the second embodiment is explained. In this second embodiment, functional items are separated into, for instance, three groups on the basis of correlation between the respective items, as shown in FIG. **11**. However, the functional items are not enveloped in the respective groups as the first embodiment, but only separated into groups. For instance, grouping is performed such that the group 1 is comprised of six functional items, the group 2 is comprised of seven functional items, and the group 3 is comprised of three functional items. Upon grouping in this way, a groups name and functional items' names are concurrently displayed.

That is, in the case of this embodiment, the menu structure itself which is shown in FIG. **11** is considered to be a list, and the list is sequentially scrolled so that all functional items are displayed. Therefore, the user is able to see both of the group name and the functional item names at a same time, and, on the basis of this two information, able to easily judge the positional difference between the portion which is now displayed and the object functional item, and to find out the object functional item with ease.

Because the group name is displayed in the case of the first embodiment too, such a point is common that the positional difference between the portion which is now displayed and the object functional item can be easily judged. However, in the case of the first embodiment, since it has hierarchical structure basically, when the group of the object functional item differs from the group of the functional item which is being displayed at present, the display must be once returned to the high hierarchy and then the group which includes the object functional item must be selected afresh, therefore, the number of steps which are needed to display the object functional item is increased. On the other hand, in the case of the second embodiment, scrolling of the list of the items is merely required because it has no hierarchical structure, and the number of steps which are needed to display the object functional item is not increased.

By the way, in the case of the second embodiment, a long side-ways liquid crystal display is needed, in order to display the menu structure shown in FIG. **11** as a single list. For that reason, in this second embodiment, a long side-ways liquid-crystal display which is able to display, for instance, 4 linesx18 columns of characters is used. In this connection, supposing that the display is performed with a small font of vertical 7 dotsxlateral 5 dots like the first embodiment, that which has the pixels of about 32 dotsx109 dots may be used.

Referring to the display layout diagrams shown in FIGS. **12** to **15**, the menu display in the second embodiment is described hereinafter.

When the menu key **6G** is pushed at first, in the portable telephone apparatus of this embodiment, the initial four items out of the group 1 are displayed as shown in FIG. **12A**. At this time, the name of the group is displayed on the upper left portion of the screen, and four functional items are displayed on the right side of the screen in regular order from the top. Besides, the cursor for indicating the portion which is being selected at present is displayed on the place of the first functional item.

When down operation of the jog dial **6J** is once performed in this state, the cursor is lowered by one line and moved to the place of the second functional item, as shown in FIG. **12B**. When down operation of the jog dial **6J** is performed once more, the cursor is further lowered by one line and moved to the place of the third functional item, as shown in FIG. **12C**.

When down operation of the jog dial **6J** is performed once more in this state, the cursor is not lowered (that is, it is caused to stay on the third line from the top of the screen) but the list of the functional items is shifted upwardly by one row, and the cursor is hereby moved to the place of the fourth functional item, as shown in FIG. **12D**. At this time, the group name is not shifted and still displayed on the upper left portion of the screen.

When down operation of the jog dial **6J** is performed once more in like manner, the functional item list is shifted upwardly by one row so that the cursor is moved to the place of the fifth item as shown in FIG. **12E**.

When down operation of the jog dial **6J** is once performed in such a state that the last item of the group 1, that is, the sixth functional item, is displayed on the fourth line of the screen as shown in FIG. **12E**, the functional item list is not shifted this time and the cursor is lowered by one row, so that the cursor is moved to the place of the sixth item, as shown in FIG. **12F**. The aim of the fact that the functional item list is not shifted and the cursor is lowered in this way at the time of the last functional item is to cause the user to know that it is the last functional item.

When down operation of the jog dial **6J** is further performed in such a state that the cursor lies in the place of the last functional item of the group 1 as shown in FIG. **12F**, the functional items of the group 2 are displayed this time as shown in FIG. **12G**. At this time, since the group has been changed, the group name which is displayed on the upper left of the screen is switched from the group 1 to the group 2. Besides, the initial four items out of the group 2 are displayed as the functional items, and the cursor is displayed on the place of the first functional item.

When down operation of the jog dial **6J** is performed sequentially in this state, the cursor is lowered one line at a time in regular order, and moved to the place of the second functional item and then to the place of the third functional item, as shown in FIGS. **13H** and **13I**.

After this, when down operation of the jog dial **6J** is performed, the cursor is moved to the place of the seventh functional item, which is the last functional item of the group 2, as shown in FIG. **13J**. At this time too, the list of the functional items is not shifted and the cursor is lowered by one line, so that the cursor is moved to the place of the seventh functional item, as with the aforementioned case.

When down operation of the jog dial **6J** is performed once more in such a state that the cursor lies in the place of the last functional item of the group 2 as shown in FIG. **13J**, the functional items of the group 3 are displayed this time as shown in FIG. **13K**. In this case too, the group name which is displayed on the upper left of the screen is switched to the group 3, and the cursor is displayed on the place of the first functional item, as with the aforementioned case.

When down operation of the jog dial **6J** is performed sequentially in the same way, the cursor is lowered one line at a time in regular order, and moved to the place of the second functional item and then to the place of the third functional item, as shown in FIGS. **13L** and **13M**.

When down operation of the jog dial **6J** is further performed in such a state that the cursor lies in the place of the last functional item of the group 3 as shown in FIG. **13M**, the display is returned again to the group 1 repeatedly, as shown in FIG. **13N**.

In contrast with this, when up operation of the jog dial **6J** is performed, the functional items are displayed in a manner

shown in FIGS. 14A to 14G and 15H to 15L. That is, when up operation of the jog dial 6J is performed in such a state that the menu key 6G has been pushed and the functional items have been first displayed (that is, such a state that the functional items of the group 1 are being displayed as shown in FIG. 14A), the functional items of the group 3 are displayed as shown in FIG. 14B. At this time, the group name of the group 3 is displayed on the upper left of the screen in like manner, and the cursor is displayed on the place of the third functional item, which is the last functional item of the group 3.

When up operation of the jog dial 6J is further performed continuously in this state, the cursor is raised one line at a time, and moved to the place of the second functional item and to the place of the first functional item in regular order, as shown in FIG. 14C and FIG. 14D.

When up operation of the jog dial 6J is further performed in such a state that the cursor lies in the place of the first functional item as shown in FIG. 14D, the functional items of the group 2 are displayed this time as shown in FIG. 14E. At this time, the group name of the group 2 is displayed on the upper left of the screen in like manner, and the cursor is displayed on the place of the seventh functional item, which is the last functional item of the group 2.

When up operation of the jog dial 6J is further performed in this state, the cursor is raised one line at a time in regular order, and moved to the place of the sixth functional item and then to the place of the fifth functional item, as shown in FIG. 14F and FIG. 14G.

When up operation of the jog dial 6J is performed in such a state that the cursor lies in the second line from the top as shown in FIG. 14G, the cursor is not raised (that is, the cursor stays at the second line) and the list of the functional items is shifted by one row, so that the cursor is moved to the place of the fourth functional item, as shown in FIG. 15H.

When up operation of the jog dial 6J is performed continuously in like manner, the functional item list is shifted downwardly in regular order so that the cursor is moved to the place of the third item and then the place of the second item, as shown in FIG. 15I and FIG. 15J.

When up operation of the jog dial 6J is performed in such a state that the first functional item of the group 2 is being displayed on the first line of the screen as shown in FIG. 15J, the functional item list is not shifted this time and the cursor is raised by one row, so that the cursor is hereby moved to the place of the first item as shown in FIG. 15K. The aim of the fact that the functional item list is not shifted and the cursor is raised in this way at the time of the first functional item is to cause the user to know that it is the first functional item.

When up operation of the jog dial 6J is performed in such a state that the cursor lies in the place of the first functional item as shown in FIG. 15K, the functional items of the group 1 are displayed again as shown in FIG. 15L. At this time, the rear four functional items out of the functional items of the group 1 are displayed, and the cursor is displayed on the place of the sixth functional item, which is the last functional item of the group 1.

In this way, in the case of this embodiment, when up or down operation of the jog dial 6J is performed, scroll-displaying of the respective functional items of the group 1 to the group 3 is performed, and so it is able to adjust the cursor to the place of the desired functional item without switching of hierarchy. When the jog dial 6J is clicked after the cursor has been adjusted to the place of the desired functional item in this way, it is able to enter the setting mode of the very functional item and to perform setting with respect to this functional item.

The display of these menu screens is performed under the display controlling of the CPU 7; the procedure of that time is described hereinafter referring to the flow chart shown in FIG. 16.

First, the menu key 6G is pushed by the user, at the step SP21 which is following to the step SP20. Accepting this, the CPU 7 goes to the following step SP22, and displays the list of the functional items of the group 1 as the menu screen. At this time, the name of the group 1 is displayed on the upper left portion of the screen as the group name, and the cursor is displayed on the place of the first functional item, which is the foremost functional item of the group 1 (see FIG. 12A or FIG. 14A).

Next, at the step SP23, up or down operation of the jog dial 6J is performed by the user. Accepting this, the CPU 7 goes to the following step SP24 and, at here, calculates the position of the cursor which is instructed with the up or down operation.

On the basis of the result of the aforementioned calculation, the portion of the functional item list which has been instructed with the up or down operation is displayed, and the cursor is displayed on that portion, at the next step SP25. As an example, if the up or down operation ranges to the item 3 of the group 2, then the functional item list is scrolled so as to display the item 3 of the group 2, and the cursor is displayed on this portion. In this case too, the name of the group which includes the functional items which are being displayed on the screen at present is displayed on the upper left of the screen.

After this, when the user clicks the jog dial 6J, entering into the setting mode of this functional item is achieved, and the procedure of the menu display advances to the step SP26 so as to end the processing.

In the above constitution, in the case of this second embodiment, when the menu key 6G is pushed, the list of the first functional item to the fourth functional item of the group 1 is displayed at first. At this time, the name of the group 1 is displayed on the upper left of the screen as the group name, and the cursor is displayed on the place of the first functional item out of the functional items which are being displayed. In this state, when the jog dial 6J is clicked after the jog dial 6J has been operated and the cursor has been adjusted to the place of the desired functional item, entering into the setting mode of the selected functional item is achieved, and setting with respect to the functional item can be performed.

By the way, in the case of this second embodiment, the functional items are separated to form some groups, and, at the time of the display the list of the functional items, the name of the group is displayed; therefore, taking the group's name as a guide, the object functional item can be easily found out.

Besides, in the case of this second embodiment, the structure is such that the functional items have been only separated to form some groups (that is, the structure is not hierarchical structure); therefore, when the functional item which is being displayed at present is different from the object functional item, the object functional item can be easily found out by only operating the jog dial 6J and scrolling the functional item list. In this connection, in the case of the hierarchical structure like the first embodiment, when the functional item which is being displayed at present is different from the object functional item, it may be obliged

to return to the high hierarchy; however, such troubles do not exist in the case of this second embodiment, and so the user's trouble is omitted and the function can be improved in addition.

According to the above constitution, owing to the fact that functional items are separated to form some groups and, at the time of the display the list of the functional items, the name of the group is displayed, the user is able to find out the object functional item taking the groups name as a guide, therefore, the object functional item can be found out with ease. In this manner, a portable telephone apparatus can be realized which is able to provide a display on which the user can easily find out the object item at the time of the display of the list of plural items.

While in the aforesaid first embodiment, the functional items have been displayed in an indented manner at the time of the display of the functional items which are the low hierarchy, and the display layout of it has been hereby differentiated from that of the display of the groups which are the high hierarchy, however, the present invention is not limited thereto and, as shown in FIGS. **17A** to **17J**, at the time of the display of the functional items the character font can be changed so as to change the shape of the displayed characters, and the display layout of it can be differentiated from that of the display of the high hierarchy. also, the size of the character font can be changed. Alternatively, for instance, a color liquid crystal display can be employed, and the display layouts of the high hierarchy and the low hierarchy can be differentiated by changing the color of the screen. To be short, if the display layouts are differentiated between the high hierarchy and the low hierarchy, it can be discriminated in which hierarchy the user is at present, and the same effects as the above can be obtained.

Further, while in the aforesaid first embodiment, the indention has been performed by two characters from the head of the line at the time of displaying the functional items which are the low hierarchy, however, the present invention is not limited thereto and the indention can be performed by one character from the head of the line, at the time of displaying the functional items. In brief, if the display positions of the items are differentiated between the high hierarchy and the low hierarchy, the same effects as the above can be obtained.

Further, while in the aforesaid first embodiment, the structure of the menu has been settled to the two-stage hierarchical structure, however, the present invention is not limited thereto and even the case where the hierarchical structure is other than such, by changing the display layout in accordance with the depth of the hierarchy, the same effects as the above can be obtained.

Further, while in the aforesaid first embodiment, the liquid crystal display apparatus can perform the display of 5 lines×16 columns, however, the present invention is not limited thereto and even the case where a liquid crystal display apparatus having the other number of lines or the other number of columns is employed, the same effects as the above can be also obtained. However, a liquid crystal display which can perform the display of three lines or more is preferable, in order to offer the user a display which is easy to recognize.

Further, while in the aforesaid second embodiment, the boundary of the group has been noticed to the user by changing the position of the cursor at the place of the first or the last functional item of the group (see FIG. **12F**, FIG. **15K**, etc.), however, the present invention is not limited thereto and the boundary of the group can be noticed by displaying a broken line, etc. for indicating the boundary of the group, as shown in FIG. **18**. Alternatively, the other line, character, sign, etc. for indicating the boundary of the group can be displayed. In brief, if the information which indicates the boundary of the group is displayed, the user can recognize the end or the beginning of the group by viewing the display.

Further, while in the aforesaid second embodiment, the group name has been displayed on the upper left side of the liquid crystal display, however, the present invention is not limited thereto and the group name can be displayed on the upper side of the functional items in the same way as the first embodiment, and, alternatively, the group name can be displayed on the other position of the screen. In brief, the display position of the group name is not limited in the present invention.

Further, while in the aforesaid second embodiment, the liquid crystal display can perform the display of 4 lines×18 columns, however, the present invention is not limited to thereto and even the case where a liquid crystal display having the other number of lines or the other number of columns is employed, the same effects as the above can be also obtained. However, in such case too, a liquid crystal display which can perform the display of three lines or more is preferable, in order to offer the user a display which is easy to recognize.

Further in the aforesaid first and second embodiments, the jog dial **6J** which has the rotating mechanism for selecting information and the click mechanism for determining selected information has been employed as the inputting means which is employed to move the cursor and to determine the item, however, the present invention is not limited thereto and various inputting means having other structure can be applied, if their operating direction for inputting movement of the cursor and the operating direction for fixing upon the selected item are different.

For instance, a so-called jog-shuttle of which rotating angle is limited within the stated angle can be employed, and a track-ball or a joy-stick which are provided with a click mechanism can be employed, and a slide-switch which is provided with a click mechanism can be employed. Employing these, a portable telephone apparatus of which number of the operation keys is little and which is handy to carry can be also realized, as with the aforesaid case.

Further, while in the aforesaid first embodiment, the display layout is changed for each hierarchy by the display controlling of the CPU **7**, however, the present invention is not limited thereto and if such a control means is provided that changes the display layout for each hierarchy when it causes the display means to display the list of the items, the same effects as the above can be obtained.

Furthermore, in the aforesaid second embodiment, the group name is displayed based on the display controlling of the CPU **7** at the time of the display of the functional item list, however, the present invention is not limited thereto and in the case of displaying the list of the plural items, if the plural items are separated into some groups, and if such a ontrol means is provided that it causes the display means to display the name of the group to which the displayed items belong, the same effects as the above can be obtained.

While the above has been described in connection with the preferred embodiments of the invention, it will be obvious to those skilled in the art that various changes and modifications may be made, therefore it is intended, to cover in the appended claims all such changes and modifications as fall within the true spirit and scope of the invention.

What is claimed is:

1. A portable communication terminal comprising:

a display for displaying a desired item selected out of a plurality of selection items which are hierarchically arranged;

a pointer for indicating said desired item out of said plurality of selection items displayed on said display, said pointer includes a rotatable disc-like member which partly sticks out from a main body of the portable communication terminal;

an instruction switch for causing an item of low hierarchy corresponding to an item indicated by said pointer to be displayed on said display; and

a data processor for changing a display layout at a time of switching by said pointer and switch to change from a display listing selection items of high hierarchy in a first format to a display listing selection items of low hierarchy in a second format, said first and second formats being recognizably different.

2. The portable communication terminal according to claim 1,

wherein said data processor includes means, at the time of switching by said pointer and switch from the display of the selection items of the high hierarchy to the display of the selection items of the low hierarchy, for simultaneously displaying information indicating the high hierarchy which is linked with said low hierarchy.

3. The portable communication terminal according to claim 1,

wherein said data processor includes means, at the time of switching by said pointer and switch from the display for the selection items of the high hierarchy to the display for the items of the low hierarchy, for displaying the character display beginning position of items displayed in said low hierarchy so as to be different from the character display beginning position of selection items which are in said high hierarchy.

4. The portable communication terminal according to claim 1,

wherein said data processor includes means for scrolling the selection items of the high hierarchy displayed on said display for every page in accordance with operations of said pointer and for scrolling plural items of the low hierarchy displayed on said display line by line.

5. The portable communication terminal according to claim 1,

wherein said data processor includes means for causing said display to display a selection item of the high hierarchy on said display together with items of the low hierarchy which are linked with said displayed selection item.

6. The portable communication terminal according to claim 5,

wherein said data processor includes means, in the case of displaying a first selection item and a second selection item which are in the high hierarchy at the same time on the display by said pointer, for displaying an indication showing a boundary of said first selection item and said second selection item.

7. The portable communication terminal according to claim 1,

wherein said data processor includes means, at the time of switching by said pointer and switch from the display for selection items of the high hierarchy to the display for items of the low hierarchy. for changing a color of the display for items of the low hierarchy relative to a color of the display for selection items of the high hierarchy.

8. The portable communication terminal according to claim 1,

wherein said data processor includes means, at the time of switching by said pointer and switch from the display for selection items of the high hierarchy to the display for items of the low hierarchy, for changing a size of a font for displaying items of the low hierarchy relative to a size a font for displaying selection items of the high hierarchy.

9. The portable communication terminal according to claim 1,

wherein said data processor includes means, at the time of switching by said pointer and switch from the display for selection items of the high hierarchy to the display for items of the low hierarchy, for changing a shape of a font for displaying items of the low hierarchy relative to a shape of a font for displaying selection items of the high hierarchy.

10. A portable communication terminal comprising:

a display for displaying a desired item selected out of a plurality of selection items which are hierarchically arranged;

a pointer for indicating said desired item out of said plurality of selection items displayed on said display;

an instruction switch for causing an item of low hierarchy corresponding to an item indicated by said pointer to be displayed on said display; and

a data processor for changing a display layout at a time of switching by said pointer and switch to change from a display of selection items of high hierarchy to a display of selection items of low hierarchy, wherein

said pointer for indicating a desired item out of said plurality of selection items displayed on said display includes a rotatable disc-like member which partly sticks out from a main body of the portable communication terminal.

11. A portable communication terminal comprising:

a display for displaying a desired item selected out of a plurality of selection items which are hierarchically arranged;

a pointer for indicating said desired item out of said plurality of selection items displayed on said display;

an instruction switch for causing an item of low hierarchy corresponding to an item indicated by said pointer to be displayed on said display; and

a data processor for changing a display layout at a time of switching by said pointer and switch to change from a display of selection items of high hierarchy to a display of selection items of low hierarchy, wherein

said pointer for indicating a desired item out of said plurality of selection items displayed on said display includes a rotatable disc-like member which partly sticks out from a main body of the portable communication terminal, and said instruction switch for is operated by pushing the disc-like member, which partly sticks out from the main body of the portable communication terminal, in a direction of the main body of portable communication terminal.

* * * * *

US005559945A

United States Patent [19]

Beaudet et al.

[11] **Patent Number: 5,559,945**

[45] **Date of Patent: Sep. 24, 1996**

[54] **DYNAMIC HIERARCHICAL SELECTION MENU**

[75] Inventors: **Phillip Beaudet**, Pickering, Canada; **Matthew Black**; **Jonathan Edwards**, both of Lexington, Ky.; **Julian Jones**, Toronto, Canada; **Eduardus A. T. Merks**; **Roger Spall**, both of North York, Canada

[73] Assignee: **International Business Machines Corporation**, Armonk, N.Y.

[21] Appl. No.: **233,160**

[22] Filed: **Apr. 25, 1994**

[30] **Foreign Application Priority Data**

May 4, 1993 [CA] Canada 2095452

[51] **Int. Cl.**[6] **G06F 3/14**

[52] **U.S. Cl.** **395/156**; 395/160; 395/157

[58] **Field of Search** 395/156, 160, 395/157, 159; 345/146, 119, 902

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,821,211	4/1989	Torres	395/156
4,920,514	4/1990	Aoki	395/156 X
5,179,653	1/1993	Fuller	395/156
5,263,174	11/1993	Layman	395/156 X
5,425,140	6/1995	Bloomfield et al.	395/156
5,463,727	10/1995	Wiggins et al.	395/156
5,485,175	1/1996	Suzuki	395/156

OTHER PUBLICATIONS

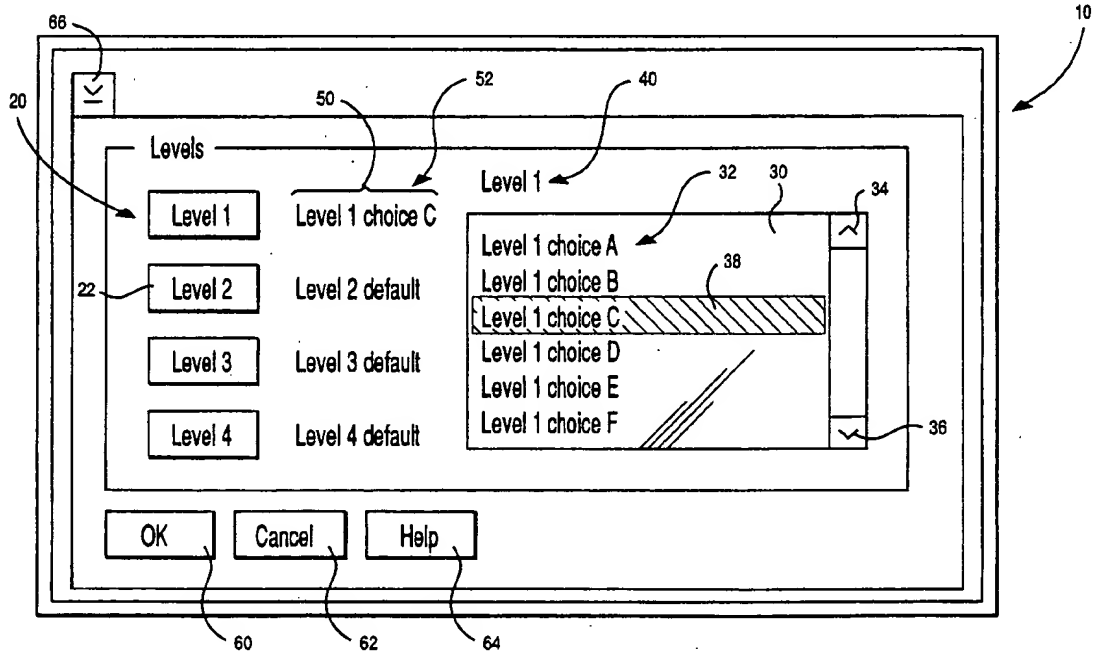
Research Disclosure No. 34698, List Boxes Within Combination Boxes or List Boxes.

Primary Examiner—Raymond J. Bayerl
Attorney, Agent, or Firm—Douglas H. Lefeve

[57] **ABSTRACT**

A control window for navigating and implementing menu selections in a multiple level hierarchical menu is provided having a first zone displaying a list of the selection levels and a second zone displaying a list of the selection options within a selection level. The second zone is refreshed with selections for another selection level in the menu hierarchy in response to a user implementation or registration of a selection option at the originally displayed selection level. Preset default settings in all selection levels permit the user to implement a "fast path" hierarchical menu selection.

20 Claims, 2 Drawing Sheets



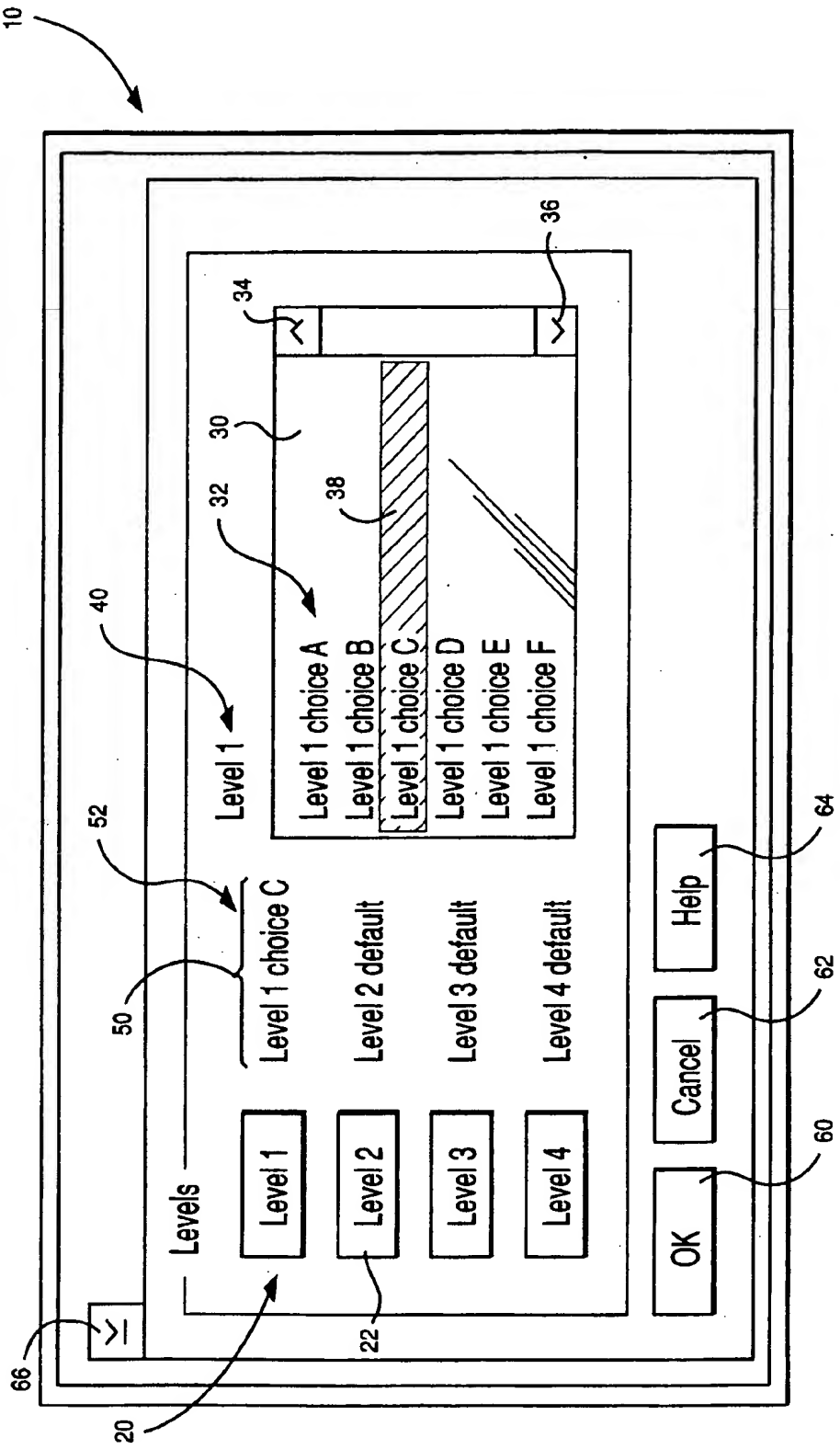
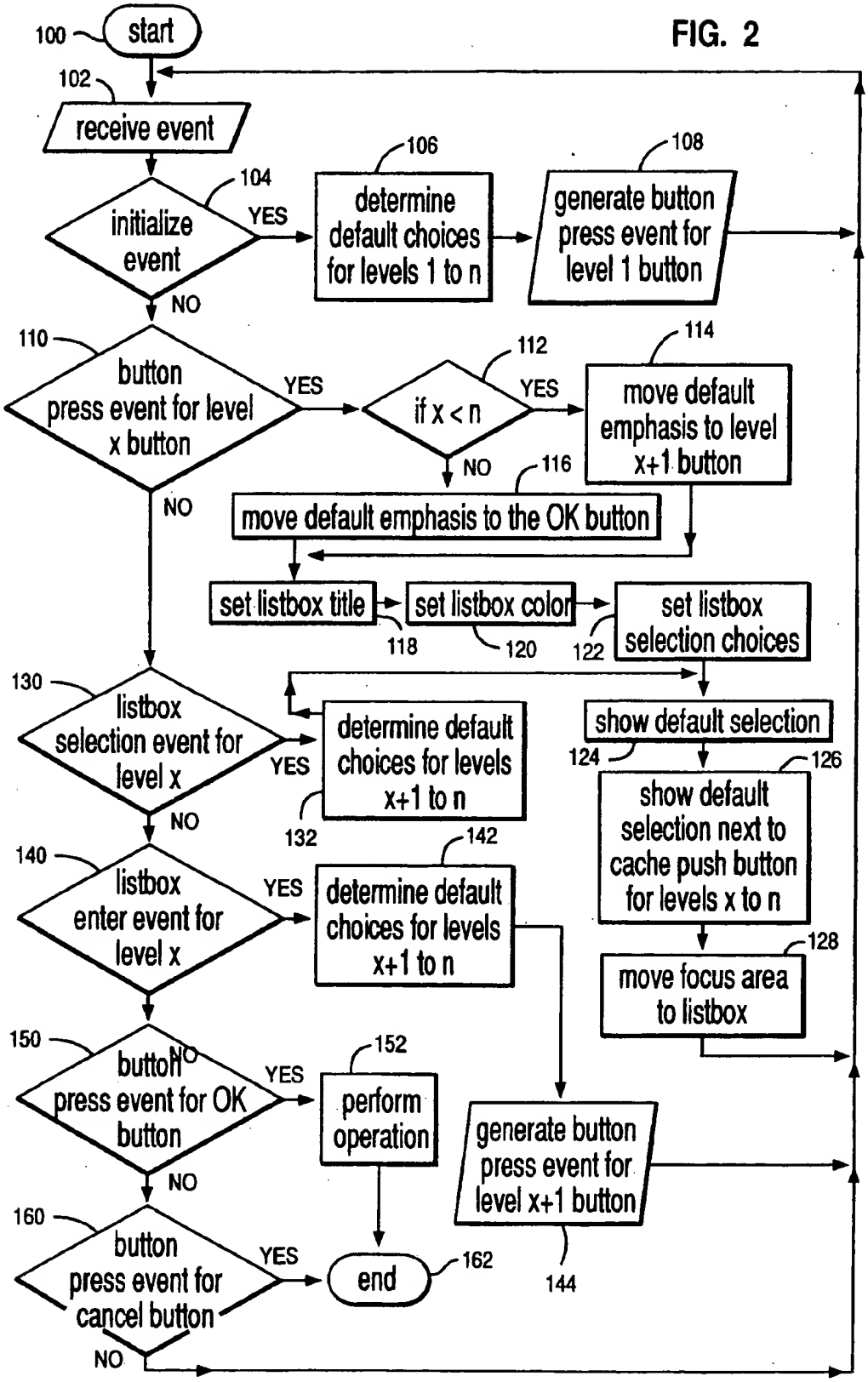

10
36
34
30
32
38
40
Level 1
Level 1 choice A
Level 1 choice B
Level 1 choice C
Level 1 choice D
Level 1 choice E
Level 1 choice F
52
50
Level 1 choice C
Level 2 default
Level 3 default
Level 4 default
Help
64
62
Cancel
60
OK
Levels
Level 1
Level 2
Level 3
Level 4
22
66
20

FIG. 1

FIG. 2


100
start
102
receive event
104
initialize event
YES
NO
106
determine default choices for levels 1 to n
108
generate button press event for level 1 button
110
button press event for level x button
YES
NO
112
if x < n
YES
NO
114
move default emphasis to level x+1 button
116
move default emphasis to the OK button
set listbox title
118
set listbox color
120
set listbox selection choices
122
130
listbox selection event for level x
YES
NO
determine default choices for levels x+1 to n
132
show default selection
124
126
show default selection next to cache push button for levels x to n
128
move focus area to listbox
140
listbox enter event for level x
YES
142
determine default choices for levels x+1 to n
150
button press event for OK button
NO
YES
152
perform operation
generate button press event for level x+1 button
144
160
button press event for cancel button
YES
NO
end
162

DYNAMIC HIERARCHICAL SELECTION MENU

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to the general area of computer user interface, and is particularly directed to a control window providing selection options and display of multiple selection levels.

2. Description of Related Art

"Menus" are the format used in computing environments for displaying to the user, in the form of prompts, the choices available to select and implement a computer task.

In order to make user interfaces, such as menus, widely useable, it is quite important that the prompts displayed are accessible to the user, that is, the user can easily recognize the nature of the prompts and work with them. Displaying obscure prompts or cluttering a menu presentation simply confuses the user and reduces the usability of the computer application.

Thus, where the choice set in a menu is too large to display at once, the menu is generally subdivided into a logically structured hierarchy that the user passes through by making selections at each level of the hierarchy and putting in place a complete set of command parameters for the computer's performance of the desired task.

For further clarity of presentation, it is usually provided that the selection of an option at one level determines the displayed range of options for subsequent (lower) levels in the menu hierarchy.

The forgoing leads to a common problem that arises in user interface design, namely the support of choice selection through the multiple levels or tiers of a hierarchical menu structure, especially where the range of choices available at one level is dependent on the selections made at a previous or higher level.

A hierarchical menu situation can arise in many different types of applications, from the systems analyst's implementation of a database search or object/class query, to the everyday user's start-up of a personal computer management program.

For example, a query action may require the specification of several (possibly four or more) query field values, where each is chosen from a mutually exclusive set of values or choices. Furthermore, selection of the first value may change the available set for the second value, and so on through to the last level or tier, creating the effect of hierarchical levels through which the user must navigate while making selections at each level that will affect the choices available at subsequent levels. The query action will be implemented only after the value for the final level has been selected.

A somewhat similar situation can be faced by the everyday PC user in accessing a simple personal computer management program, such as a word processing program. The user may be faced with a series of menus each displaying several options, where the choice in each menu can determine the range of choices available in the menu at the next level.

In all of these applications, one traditional format for presenting the choices to the user is as a cascade of menus, each menu displaying only the choices available at that level. Once the choice is made, the menu and its selection disappear from view to be replaced by the menu listing the possible selections for the next level in descending through the levels of the menu selection. Thus, the user has no visual record of menu selections from previous higher levels and cannot easily alter an inappropriate earlier menu selection without passing through each earlier menu in the cascade in order to locate the selection now sought to be corrected.

Proposed methods for simplifying hierarchical selections and providing the user with ready access to earlier selections employ multiple listboxes. A single listbox is provided for every level of menu selection listing all possible options. In one format, the listboxes are "tiled" in descending order from one side to the other on a laterally scrolled display. However, as all permutations of menu level selections must be displayed simultaneously on the multiple listboxes, the number of menu level selections or the complexity of each menu level is necessarily determined by the display size.

In a second format, the multiple listboxes are displayed in offset overlay (as multiple "windows"), permitting the user to bring the list box of an earlier selection level to the foreground (by clicking the mouse on it) for review or change. Again, however, screen size, along with the desire to avoid undue clutter in presentation, limits the hierarchical depth and individual list box length (or range of options) for which this style of interface is viable.

A similar approach to that of multiple listboxes is taken in U.S. Pat. No. 5,179,653 to Fuller. In this menu system, pushbuttons for three selection levels of a menu system are displayed simultaneously around the periphery of a graphics data work area on a display screen. Again, the display size is a limiting factor in the number of menu selection levels that may be presented, and of the complexity at each level. In Fuller, it is specifically disclosed that the labelling on the pushbuttons at the second and third descending levels are generic in nature to increase the range of options. However, use of generic labelling limits the usability of this menu display to expert users.

SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a control mechanism for implementing a user interface to display, in a single screen format, a hierarchical or multiple level selection path that is easily accessible by the novice user.

It is also an object of the present invention to provide a multiple level selection menu with a fast path for the expert user.

It is a further object of the present invention to provide on a single screen, multi-level menu display with dynamically changing values to reflect selections entered by the user.

According to the foregoing objects, the present invention offers a simple yet elegant approach to the problem of presenting hierarchical menu selections to the computer user by providing a single selection screen which combines a finite number of pushbuttons with text fields and a single, dynamically changing listbox. This arrangement increases the user's comprehension and comfort in passing through the cascading levels of selection options in a menu hierarchy.

In one aspect, in a computer system having a display screen, the invention provides a user interface mechanism for hierarchical menu selections. The user interface comprises a zoned menu control window that includes a first zone displaying indicators representing a plurality of selection levels in a menu hierarchy, and a second zone having contents comprising a displayed list of selection options for

one of the selection levels in the menu hierarchy. The interface also includes means for entering an option selected from the displayed list of options, means for refreshing the contents of the second zone by replacing the displayed list of selection options with a second list of selection options for a different selection level in the menu hierarchy in response to entry of an option selected from said displayed list of selection options, and registration means for implementing n computer task on selection of options for all selection levels in the menu hierarchy.

Preferably, the means for refreshing the contents of the second zone replaces the displayed list of selection options with a second list of selection options for an inferior selection level in the menu hierarchy.

Preferably, the indicators in the first zone consist of a plurality of pushbuttons, each pushbutton correlating to one selection level within the menu hierarchy, and the second zone consists of a listbox.

Preferably, the control window also includes a text field adjacent the pushbuttons stating the selection effected at each level, thereby providing a visual record to the user of the selections made at each level of the menu hierarchy.

In another aspect, the present invention is also directed to a method for assembling and displaying a hierarchical selection menu on a computer display screen. The method comprises the computer implemented steps of creating a zoned menu control window, displaying in a first zone of the control window, a number of indicators, each of which corresponds to a separate selection level in the menu hierarchy, and displaying in a second zone of the control window, a list of selection options for one of the selection levels in the menu hierarchy. In response to entry of an option from the displayed list of selection options, the displayed list of selection options in the second zone is replaced with a second list of selection options for a different selection level in the menu hierarchy.

The present invention is also directed to a method of assembling and displaying a hierarchical selection menu on a computer screen. This method comprises the computer implemented steps of creating a menu control window having multiple zones, displaying in a first zone in the menu control window, a plurality of pushbuttons each of which represents a separate level in the menu hierarchy, displaying within a second zone in the menu control window, a list of selection options for the highest level not yet selected in the menu hierarchy, and, in response to a pushbutton level selection, refreshing the list of displayed selection options with a list of available selection options for the next highest level not yet selected in the menu hierarchy.

In a further aspect, the present invention is directed to a fast process for implementing hierarchical menu selections in which default selection values are preset and displayed to the user for registration as option selections to implement a computer task.

BRIEF DESCRIPTION OF DRAWINGS

Embodiments of the invention will now be described in detail in association with the accompanying drawings, in which:

FIG. **1** is a pictorial view of a control window for effecting and displaying multiple level choices: and

FIG. **2** is a flow chart showing the event steps for computer implementation of a dynamic user interface for a hierarchical selection menu according to the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Referring to FIG. **1**, a hierarchical selection menu is presented in the form of a control window, generally designated by the numeral **10**. As discussed hereafter, the control window **10** contains multiple zones or fields representative of the hierarchical levels and selection options at each level that comprise the preferred embodiment of this application.

A first zone **20** in the control window **10** displays a vertical array of pushbutton indicators **22**. Each pushbutton **22** corresponds to and is representative of a separate selection level in the menu hierarchy. For ease of use and understanding, these pushbuttons are ranked in descending order from superior to inferior selection levels, that is, the pushbutton identified as "Level 1" in the control window **10** represents the primary or highest selection level in the menu hierarchy, the pushbutton identified as "Level 2" represents the second level after the first in the menu hierarchy, and so on down to the lowest or most inferior pushbutton/menu selection level.

Opposite the first zone **20**, a second zone in the control box **10** displays a list box **30** containing a list of the selection options (designated by numeral **32**) available at the current selection level under consideration. This current selection level will normally be the highest level in the menu hierarchy at which no selection has been made by the computer user. However, the invention also provides for special situations where the user's selections are entered out of hierarchical order. For example, the provision of preset default values in some or all of the hierarchy levels permits the user to selectively enter inferior or lower selection levels in the menu hierarchy. Also, the user is permitted to backtrack in the menu hierarchy to change a selection made at an earlier level in the hierarchy. A title field or bar **40** is superimposed over the list box **30** and displays the name of the selection level corresponding to the options displayed in the list box **30**, in order to clearly direct the user in the selection level being implemented.

Intermediate the first zone **20** and the list box **30** is a text field **50** displaying a line of text **52** adjacent each pushbutton **22**, providing the user with a visual record of the current selection of options throughout the menu hierarchy represented in the control window **10**.

In the preferred embodiment of the invention shown in FIG. **1**, a focus area in the form of a contrast bar **38** is provided in listbox **30** highlighting the current option selected for the particular level "x" under consideration. On entering a selection level, the focus area is automatically on the listbox **30** which has repainted itself with the list of selection options for that level, and the contrast bar **38** highlights the default value calculated and currently selected for that level. The contrast bar may be moved up and down the list **32** in the list box **30** by simply pressing the terminal cursor arrow keys up or down to highlight each selection option until the desired option is located.

The location of the contrast bar **38** on a list item **32** in the listbox **30** has a direct effect on the contents of the text field **50**. As will be described in greater detail below, the text line **52** displayed adjacent the pushbutton **22** of the level currently displayed in the listbox changes to reflect the selection option highlighted by the contrast bar **38**. In addition, the default choices of the inferior selection levels are automatically re-calculated and displayed as the text lines **52** adjacent their respective pushbuttons **22**.

When the contrast bar **38** is highlighting the user's preferred option, the user can enter the selection by known

means such as pressing the "ENTER" key or double-clicking the mouse.

As a further visual aid to the user, unique colours may be used to identify each selection level in the menu hierarchy. Each colour could appear on the pushbutton **22** associated with the levels in the menu hierarchy on initialization of the menu control window, or, alternatively, could only appear on each pushbutton once the user has initialized an event for its associated selection level. In one embodiment, when colour display is used as indicative of selection level, the text line **52** adjacent a pushbutton would be implemented in the same colour, as well as the background colour of the list box **30** at the time the list of selection options for a particular level is displayed there. This would maximize the visual correlation to the user between the three main fields in the control window **10**.

In addition, a few standard user interface features that will be readily recognizable to the ordinary computer user are provided for common functions, as discussed below.

As shown in blocks **100** and **102** in FIG. **2**, after starting and initializing the system, the computer receives an event signal generated by a computer user action, such as registration of a keyboard action or mouse movement. On first entry to the menu hierarchy, the event signal would generally include a system load of the menu hierarchy for the specific application or program selected for implementation by the computer user.

On receiving the event, the computer determines if the event must be initialized (block **104**). Initialization of the control window of FIG. **1** involves the determination and assignment of default values for all levels (n) of the menu hierarchy **106**, and the display of selection options for the primary level (Level **1**) of the menu hierarchy in the list box **30** of FIG. **1** and the title for Level **1** in the title bar **40**, before generating the "button press event" (access to a selection level in the menu hierarchy) for the pushbutton representing the primary or first level in the menu hierarchy **108** (FIG. **2**). The generation of this "button press event" becomes the new initialized event received by the computer.

In the preferred embodiment, on initiating the control window **10**, default values for all of the levels in the menu hierarchy are automatically determined and entered as preset values in the menu hierarchy (block **106** in FIG. **2**). These predetermined default values could either be set values in the system, or could be the user's own selections retained from an earlier implementation of the selection control window.

Although default values are predetermined in the preferred embodiment, the settings could be programmable to permit the expert user to tailor the system for particular applications.

The event initialization of box **106** in FIG. **2** will occur only once, on initialization of the control window **10** display shown in FIG. **1**. On receiving the "button press event" for Level **1** (block **108**), the computer's determination that no further event need be initialized (block **104**) will initiate display of the control window.

On initiation of the menu control window **10** display, the pre-set default selections appear as the text descriptions **52** in the text field **50** beside each corresponding pushbutton **22** in the first zone **20**, including that for the Level **1** pushbutton in the menu hierarchy.

Where colour coding is used, the common colours for the pushbutton **22** and text line **52** at each selection level would also be implemented on initiation of the control window **10**, and the colour coding associated with Level **1** would then be shown as the background colour for the list box **30** containing the selection options for Level **1**.

On opening the control window **10**, the contrast bar **38** is set on the default setting for Level **1** in the listbox **30** (not shown in FIG. **1**).

At this point, the computer user can determine whether to maintain the default selection provided for Level **1** or to substitute an alternate option for the default selection. The user can maintain the default selection and pass through to the next selection level simply by entering the selection using known means such as pressing the "ENTER" terminal key action or double-clicking the mouse. This selection entry constitutes the "button press event" for the next (inferior) selection level in the menu hierarchy (block **110**).

As shown at block **112** of FIG. **2**, entry of the "button press event" at any selection level causes the computer to verify whether this selection level is the last level in the menu hierarchy. Where the computer identifies the selected level as the last selection level in the menu hierarchy (block **116**), the default emphasis in the control window is shifted to a task implementation indicator. In the illustrated embodiment, the task implementation indicator is the "OK" button **60** (in FIG. **1**), which is automatically activated as the conduit to enable the user to register the command to implement the computer task established by the options currently selected in the menu hierarchy (blocks **150** and **152** in FIG. **2**).

To consider other selection options at the displayed selection level, the user can move the contrast bar **38** up and down the list of selections in the listbox **30** by using the arrow keys, or possibly by locating the mouse cursor and clicking the mouse on different options.

Whenever the contrast bar **38** moves to a different option in the selection list, the computer recognizes a "listbox selection event" (block **130**), and changes the text field **50** to show the default selections for inferior levels of the currently highlighted option. The new default selections are determined through implementation of the computer generated steps illustrated in blocks **132**, **124**, **126** and **128** of FIG. **2**, namely, determination of the default choices for all inferior levels to the level currently listed in the listbox, and repainting the text field to display those new default selections while moving the focus area back to the listbox.

It is this automatic change of inferior selection options in the invention that allows a user to return to a higher menu selection level at any time. The user can press the "ENTER" key or click the mouse on the pushbutton for a desired level in order to generate a "button press event" to implement the computer steps for that level (block **110**), the computer assuming selection entry for the immediately preceding (superior) level.

Where, following a "button press event" (block **110**), the computer identifies the selected level as being a level superior to the last selection level in the menu hierarchy (block **112**), the default emphasis is moved to the next inferior selection level in the menu hierarchy (block **114**), and the contents of the second zone are refreshed with selection information for the new selection level. The steps implemented by the computer to refresh the contents of the second zone are set out in FIG. **2**, as:

1. replacing the listbox title (in title field **40** of FIG. **1**) with the title of the new selection level (block **118**);
2. changing the background colour of the listbox **30** to the colour shown on the pushbutton associated with the new selection level (block **120**); and
3. replacing the list of selection options in the listbox with a new list of selection options available for the new selection level (block **122**).

After this the remaining steps illustrated in blocks **124**, **126** and **128** are implemented, as discussed above. Thus, immediately on fixing the contents of the listbox **30**, the contrast bar **38** highlights the selection option corresponding to the default value identified in text line **52** for the new selection level (box **124** in FIG. **2**), determines and enters the default values in text field **50** for inferior levels in the menu hierarchy (block **126** in FIG. **2**), and moves the focus area to the listbox **30** (block **128**).

When the user decides on a selection option for the level "x" currently displayed, the selection can be entered as a "listbox enter event" for the level by double-clicking the mouse or pressing the "ENTER" key when the contrast bar **38** is located on the selection (block **140**). This selection causes the default selection values for all inferior selection levels to be determined (block **142**) and generation of the "button press event" for the next inferior level (block **144**), before returning to the loop for initializing and displaying the new selection level following the steps discussed above.

The list of selection options for Level **1** displayed in the listbox **30** will naturally be the broadest available for implementing the application. However, the contents of option lists for subsequent levels may vary significantly as the available range of selection options is determined by selections made at superior levels.

Although the listings **32** of selections in the listbox **30** are presented in FIG. **1** as generic listing labels, the text for each listing within the list box **30** would normally be descriptive to aid, as much as possible, the novice user as well as the expert user exploring complex applications. Further, "PAGE UP" and "PAGE DOWN" scroll bars **34** and **36** are provided in the frame **35** of the list box to increase the usable display area within the list box **30**, and although not shown, it would be obvious to one skilled in tile art to include one or more horizontal scroll bars for longer textual descriptions.

Similarly, the title in the title field **40** would contain a descriptive, rather than generic, title for the selection level. In one format, the title would reflect the wording of the correlated pushbutton for that level.

Since in the preferred embodiment, on initiating the control window **10**, the selections in the menu hierarchy are automatically preset at default values, a fast path or process is provided for implementing computer tasks using the preset selections and bypassing the selection levels.

In addition, where the default settings are also programmable, the expert user is provided with a means for implementing a customized fast path for preparing a query or entering all application by simply registering all selection levels in the menu hierarchy using the pre-programmed selection values to bypass the selection levels.

In either case, the fast path may be implemented by the user immediately on initiation of the control window **10**, by moving the focus area or mouse cursor to the "OK" button **60** in FIG. **1**, and clicking the mouse or pressing the terminal keyboard "ENTER" key as the "button press event" to register the selections for all levels and initiate the computer s performance of the defined operation (blocks **150** and **152**, FIG. **2**).

The preferred embodiment for this invention has been illustrated in FIG. **1** as containing four nested selection levels. However, it will be obvious to one skilled in the art that the procedure illustrated in FIG. **2** could be implemented with a substantially deeper hierarchy, the provision of more levels of pushbuttons **22** in the control window **10** being made possible either by a rearrangement of the screen display, or through utilizing a scroll for off-screen levels. It should be recognized, however, that even four levels of hierarchical menu selections can represent an extensive and very detailed set of selections, and should be sufficient, with very little modification, for most applications.

In addition to the foregoing and as illustrated in FIG. **1**, standard 'HELP' and 'CANCEL' pushbuttons **62** and **64** are provided. As is known, the 'HELP' pushbutton enables the user to access information on the functioning of the option selected (in this case, information would be provided on the multi-level menu selector itself), and the 'CANCEL' pushbutton allows the user to cleanly exit the function (block **160** in FIG. **2**).

While the invention has been particularly shown and described with reference to a preferred embodiment, those skilled in the art will understand that various changes in form and detail may be made without departing from the spirit and scope of the invention.

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. In a computer system having a display screen, a user interface mechanism for hierarchical menu selections comprising:

a zoned menu control window including a first zone displaying indicators representing a plurality of selection levels in a menu hierarchy, and a second zone having contents comprising a displayed list of selection options for one of said selection levels;

means for entering an option selected from said displayed list of selection options;

means for refreshing the contents of the second zone by replacing the displayed list of selection options with a second list of selection options for a different selection level in response to entry of an option selected from said displayed list of selection options; and

registration means for implementing a computer task on selection of options for all selection levels in the menu hierarchy.

2. A computer system, according to claim **1**, wherein the means for refreshing the contents of the second zone comprises replacing the displayed list of selection options with a second list of selection options for an inferior selection level in the menu hierarchy.

3. A computer system, according to claim **1**, further comprising:

means for verifying inferior selection levels in the menu hierarchy in response to entry of an option selected from said displayed list of selection options;

a task implementation indicator linkable to said registration means; and

means for linking said task implementation indicator and registration means in response to a failure to verify inferior selection levels in the menu hierarchy, whereby to permit engagement of the registration means through the task implementation indicator.

4. A computer system, according to claim **1**, wherein the indicators displayed in the first zone comprise a plurality of pushbuttons, each pushbutton correlating to a separate selection level in the menu hierarchy.

5. A computer system, according to claim **4**, wherein the pushbuttons are ranked from superior to inferior levels in the menu hierarchy.

6. A computer system, according to claim **1**, wherein the second field comprises a scrolled listbox.

7. A computer system, according to claim **6**, wherein the second zone further comprises a title field containing a displayed title identifying said one selection level, and wherein the means for refreshing the contents of the second

02/03/2003, EAST Version: 1.03.0002

zone replaces the displayed title with a second title for said different selection level.

8. A computer system, according to claim 1, wherein the menu control window further comprises a third zone containing a text field adapted to display the options selected for each selection level in the menu hierarchy.

9. A computer system, according to claim 1, further comprising means for setting default selection values in several selection levels of the menu hierarchy.

10. A computer system, according to claim 9, wherein the means for setting default selection values sets default selection values for all levels in the menu hierarchy in response to initiation of the menu control window.

11. A computer system, according to claim 9, wherein the means for setting default selection values are programmable.

12. A computer system, according to claim 9, wherein the second zone further comprises a movable focus bar for highlighting a single selection option in the displayed list, the focus bar being linked to said means for entering an option, whereby to permit entry of a highlighted selection option.

13. A computer system, according to claim 12, wherein the means for setting default selection values sets default selection values for all inferior levels in the menu hierarchy in response to movement of the focus bar on the displayed list of selection options.

14. A computer system, according to claim 13, wherein the control window further comprises a third zone containing a text field adapted to display: option selections entered for superior levels; a highlighted selection option from the displayed list of selection options; and the default selection values for all inferior levels in the menu hierarchy.

15. A method of assembling and displaying a hierarchical selection menu on a computer display screen, comprising the computer implemented steps of:

creating a menu control window having at least a first and a second zone;

displaying in the first zone of the control window, a plurality of indicators, each indicator representing a separate selection level in a menu hierarchy;

displaying in the second zone of the control window, a list of selection options for one of said selection levels in the menu hierarchy; and

in response to entry of an option from the displayed list of selection options, replacing the displayed list of selection options in the second zone with a second list of selection options for a different selection level in the menu hierarchy.

16. A method, according to claim 15, further comprising, in response to entry of an option from the displayed list of selection options, the step of verifying inferior selection levels in the menu hierarchy, and displaying as the second list in the second zone, a list of selection options for an inferior selection level.

17. A method, according to claim 15, further comprising the step of displaying options previously selected for any selection level in the menu hierarchy.

18. A method of assembling and displaying a hierarchical listing of menu selections made by a computer user, said method comprising:

displaying a menu control window having multiple zones;

displaying within a first zone in said menu control window, a list of levels of the menu hierarchy;

displaying within a second zone in said menu control window, a list of selection options for a highest level not yet selected in the menu hierarchy;

displaying options previously selected for any selection level in the menu hierarchy; and

refreshing the second zone with a new listing of selection options from all immediately inferior selection level in the menu hierarchy in response to a user selection from the second zone.

19. A method, according to claim 18, further comprising implementing a computer task in response to user registration of entry of all selection levels in the menu hierarchy.

20. A fast process for implementing selections through a menu hierarchy, comprising the computer implemented steps of:

setting default selection values for all selection levels in the menu hierarchy;

creating a zoned menu control window; displaying in a first zone of said control window a plurality of indicators, each indicator setting forth the default selection value for a separate selection level in the menu hierarchy;

setting a movable default emphasis on one superior selection level having selection options;

displaying in a second zone a list of selection options for a selection level subject to the default emphasis; and

implementing a computer task in response to user registration of selection of options for all selection levels in the menu hierarchy.

* * * * *

US006208341B1

(12) **United States Patent**
van Ee et al.

(10) **Patent No.: US 6,208,341 B1**
(45) **Date of Patent: Mar. 27, 2001**

(54) **GUI OF REMOTE CONTROL FACILITATES USER-FRIENDLY EDITING OF MACROS**

(75) Inventors: **Jan van Ee**, San Jose, CA (US); **Paul S. Moore**; **Guy J. Roberts**, both of Eindhoven (NL)

(73) Assignee: **U. S. Philips Corporation**, New York, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/129,300**

(22) Filed: **Aug. 5, 1998**

(51) **Int. Cl.**[7] **G06F 3/00**

(52) **U.S. Cl.** **345/339**; 345/326; 345/173; 345/352; 345/335; 345/327

(58) **Field of Search** 345/970, 326, 345/339, 173, 168, 968, 352, 335, 356, 327

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,231,381		7/1993	Duwaer	345/174
5,402,151		3/1995	Duwaer	345/173
5,410,326	*	4/1995	Goldstein	348/734
5,734,853	*	3/1998	Hendricks et al.	345/352
5,745,116	*	4/1998	Pisutha-Arnond	345/358
5,767,458		6/1998	Koolen	178/18.01
5,767,919	*	6/1998	Lee et al.	348/569
5,777,607		7/1998	Koolen	345/174
5,822,123	*	10/1998	Davis et al.	348/564
5,898,398	*	4/1999	Kumai	341/176

OTHER PUBLICATIONS

Software Upgrade Booklet for the DVX8000 Multimedia Home Theater of Philips Electronics, pp. 25–31.
User Guide 206W851260 for the RC2000MkII model, pp. 22–32.
Nov. 1996 issue, pp.40–42 of the magazine "Home Theater".
Winter 1996 issue, pp. 123–126, of the magazine "Home Theater".

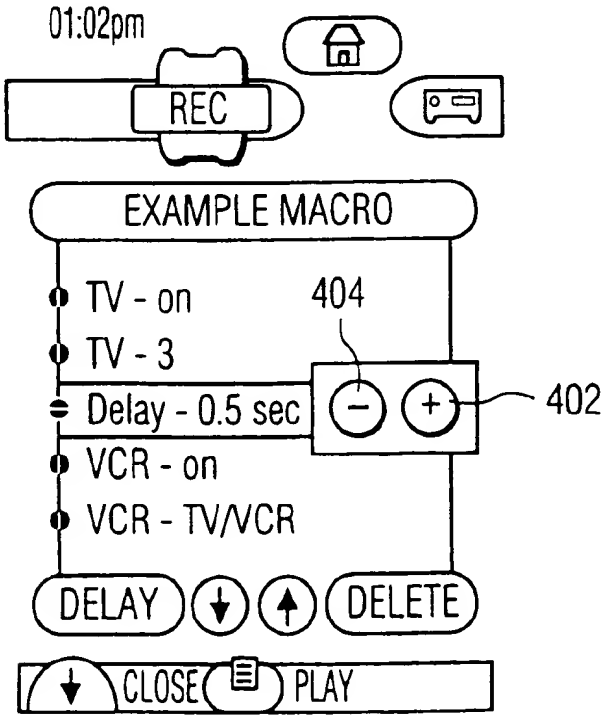
* cited by examiner

Primary Examiner—Raymond J. Bayerl
Assistant Examiner—Cuong T. Thai
(74) *Attorney, Agent, or Firm*—Peter Verdonk

(57) **ABSTRACT**

A remote control device for a home theater has a macro creation/editing mode with authoring tools on the remote's GUI. One of the editing tools lets the user move a selected macro step visibly up or down the list of steps on the GUI.

5 Claims, 4 Drawing Sheets



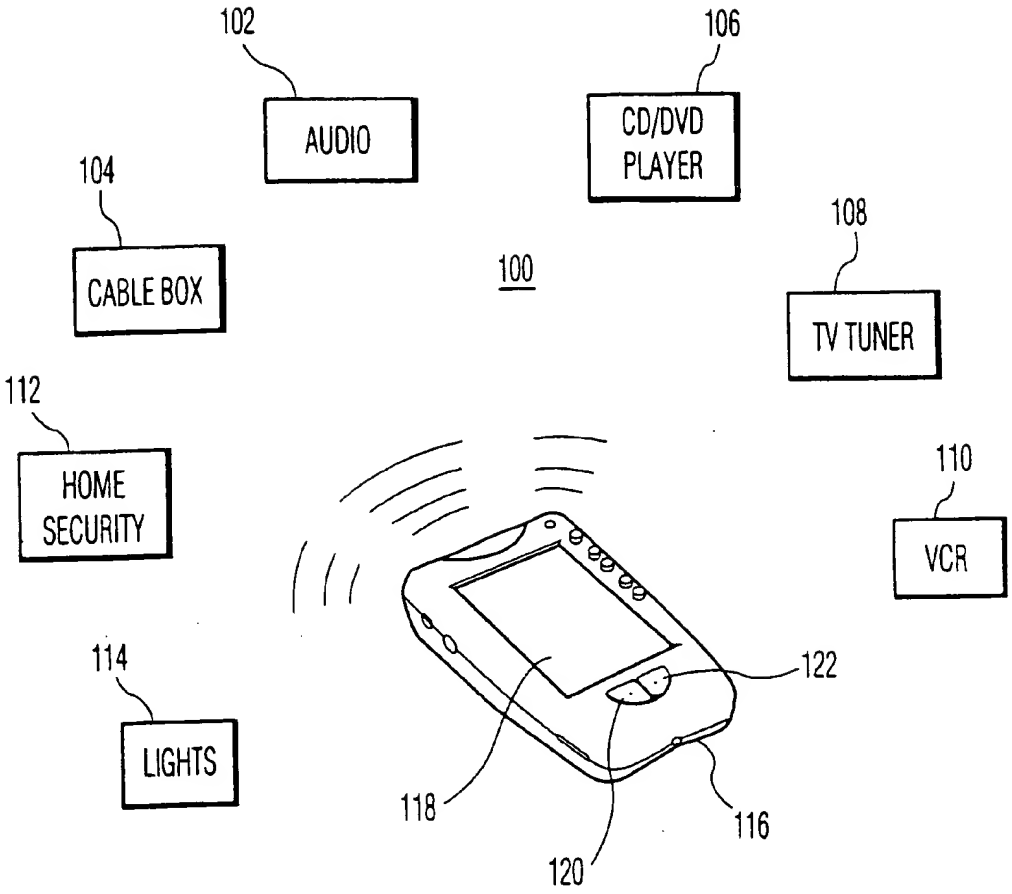

102
AUDIO
106
CD/DVD PLAYER
104
CABLE BOX
100
108
TV TUNER
112
HOME SECURITY
110
VCR
114
LIGHTS
122
118
116
120

FIG. 1

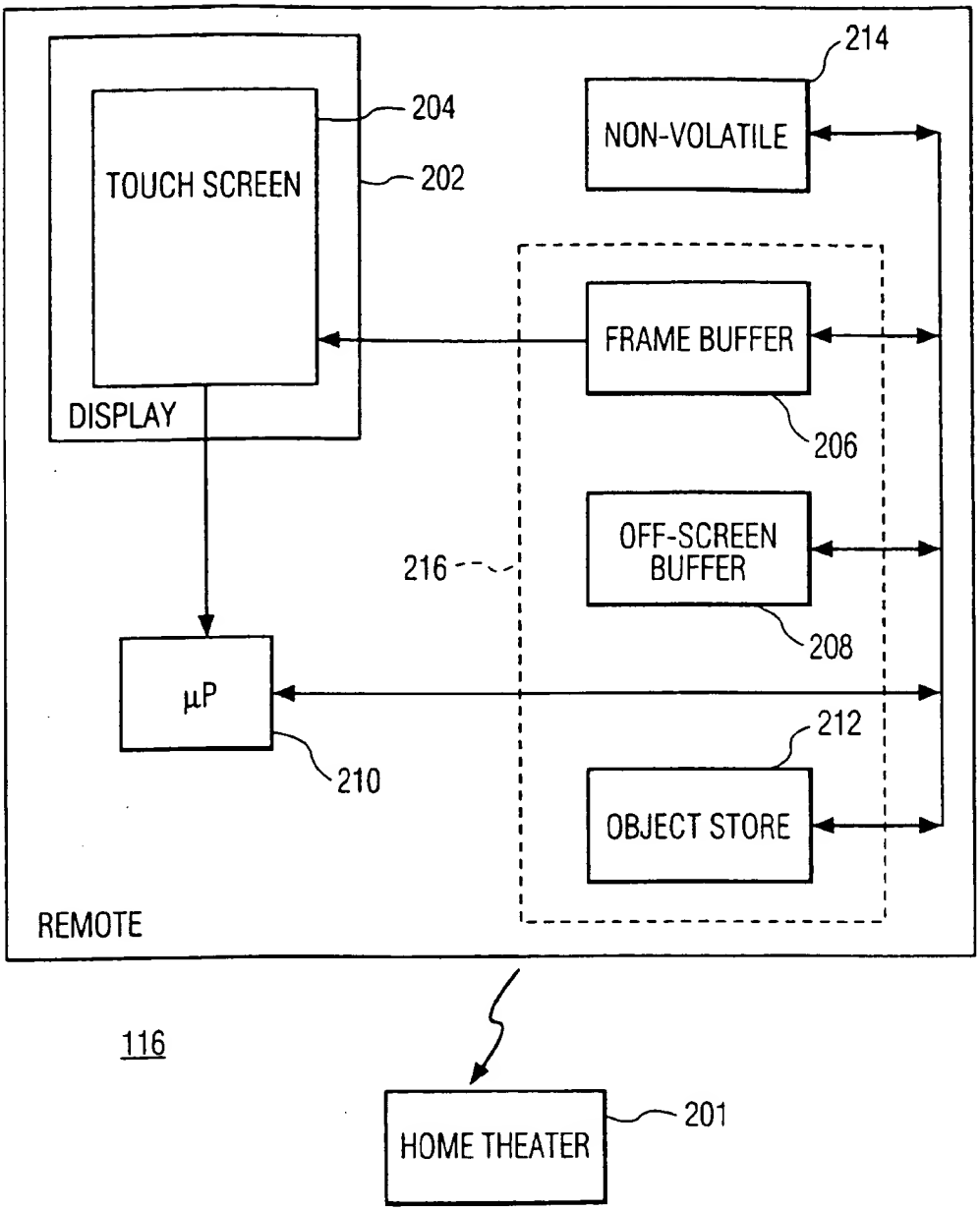

214
NON-VOLATILE
204
TOUCH SCREEN
202
FRAME BUFFER
DISPLAY
206
OFF-SCREEN BUFFER
216
208
μP
212
210
OBJECT STORE
REMOTE
116
HOME THEATER
201

FIG. 2

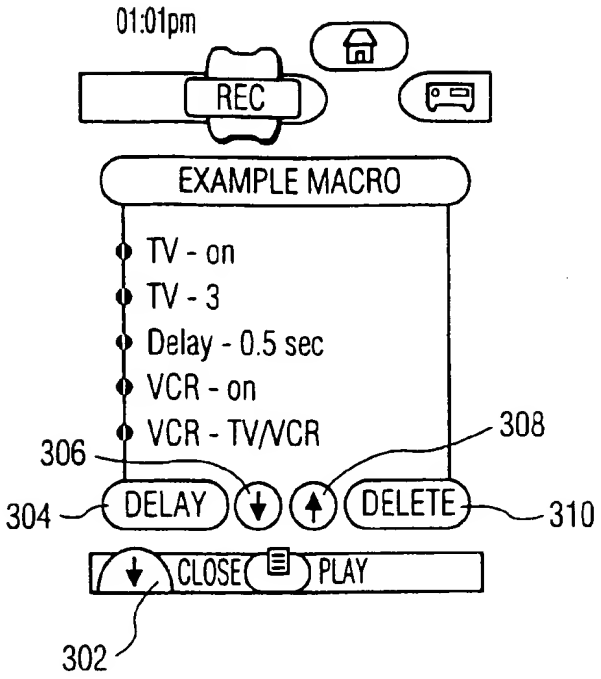

01:01pm
REC
EXAMPLE MACRO
TV - on
TV - 3
Delay - 0.5 sec
VCR - on
VCR - TV/VCR
306
308
304
DELAY
DELETE
310
CLOSE
PLAY
302

FIG. 3

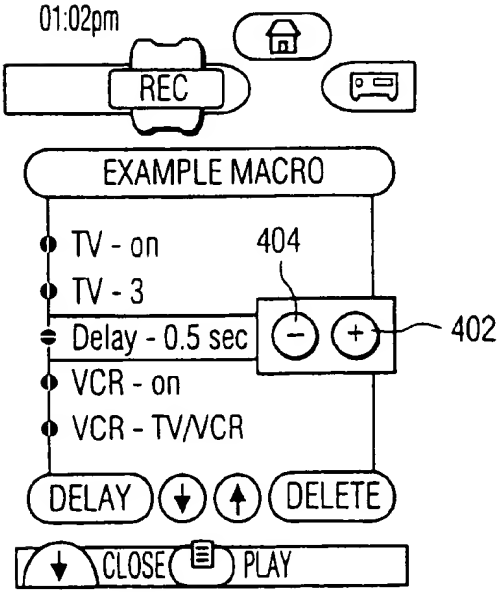

01:02pm
REC
EXAMPLE MACRO
TV - on
TV - 3
404
Delay - 0.5 sec
402
VCR - on
VCR - TV/VCR
DELAY
DELETE
CLOSE
PLAY

FIG. 4

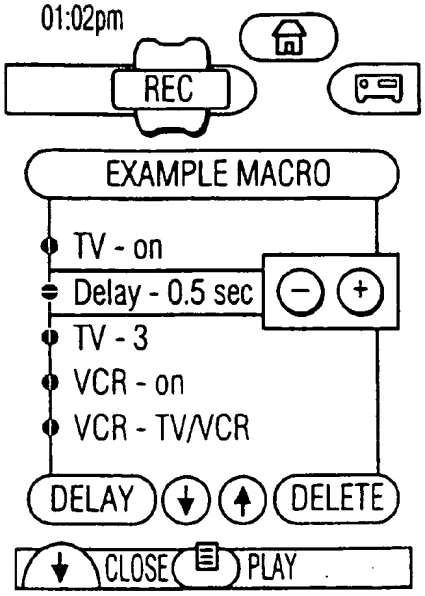

01:02pm
REC
EXAMPLE MACRO
TV - on
Delay - 0.5 sec
TV - 3
VCR - on
VCR - TV/VCR
DELAY
DELETE
CLOSE
PLAY

FIG. 5

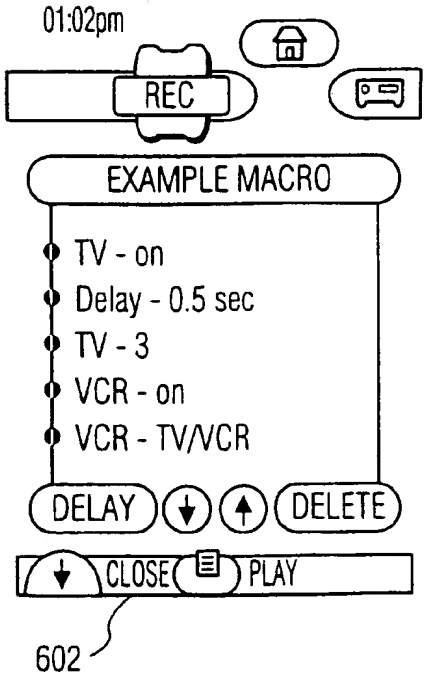

01:02pm
REC
EXAMPLE MACRO
TV - on
Delay - 0.5 sec
TV - 3
VCR - on
VCR - TV/VCR
DELAY
DELETE
CLOSE
PLAY
602

FIG. 6

GUI OF REMOTE CONTROL FACILITATES USER-FRIENDLY EDITING OF MACROS

FIELD OF THE INVENTION

The invention relates to a handheld remote control device for control of electronic equipment, in particular for control of home theater equipment. The device has a program mode that enables the user to create macro's.

BACKGROUND ART

Examples of a state of the art programmable remote control device are the RC2000 and RC2000MkII, both of Marantz, a subsidiary of Philips Electronics. These devices have the ability to learn commands for typical control functions, and to associate a respective one of the device's buttons with a respective one of the commands learned. The devices can also be programmed to store macros. A macro is a string of commands that describe a series of specific steps to be carried out in sequence. A macro is assigned to a particular button. After the device is put into the macro-creation mode, the user must enter the sequence of steps to be carried out under the macro. Entering a step corresponds to pressing one of the buttons that have already been programmed for a single command. The device has a GUI that gives the user visual feedback during programming. A GUI is an interactive user-interface with graphical representations.

For more information on the operation of the Marantz device see, e.g., the Software Upgrade Booklet for the DVX8000 Multimedia Home Theater of Philips Electronics, pages 25–31, and the User Guide 206W851260 for the RC2000MkII model, pages 22–32, herein incorporated by reference.

For more information on programming of universal remote controls see, for example, U.S. patent application Ser. No. 08/907,284, now allowed, of Philips Electronics, and herein incorporated by reference.

OBJECT OF THE INVENTION

The user can program a macro on the known devices and store it under any of four hard keys. A macro hard key can be programmed to store up to 20 steps. Revising of a programmed macro can be done in several ways: by over-writing one or more steps already entered, inserting a step at a specific position in the sequence, or deleting a step entered previously. The Marantz RC2000 remote control device has been welcomed enthusiastically in the home theater world. See, for example, the November 1996 issue. pp.40–42, and the winter 1996 issue, pp. 123–126, of the magazine "Home Theater". However, programming the known devices is not that simple and requires an effort well beyond that what the average consumer is willing to spend. An object of the invention is therefore to improve upon the user-friendliness of the known universal remote control devices, especially those for operating a consumer environment such as a home theater.

SUMMARY OF THE INVENTION

To this end, the invention provides a remote control device for control of electronic equipment. The remote has a display for display of a GUI with a plurality of user-selectable options. The remote enables the user to program a macro. The macro comprises a sequence of multiple ones of the control options of the GUI. The GUI enables the user to view the sequence as a scrollable list of steps. The GUI

enables the user to edit the list by selecting at least one specific one of the steps and moving it up or down the list.

In this manner, the user can, if so desired, create any permutation of the sequence in an intuitive fashion. Preferably, the remote of the invention has the control options organized in a hierarchy of panels for individual display on the GUI. The user can program a macro that has one or more steps that lets the GUI jump to one or more specific ones of the panels during execution of the macro. This approach lets the user intervene with the macro execution or manually add steps during execution of an individual macro. However, it may be more convenient to have the jump to a specific panel as the last step in the macro. Preferably, the remote has a touch screen for enabling user-selection of the options. All major user control functions can thus be presented through graphical representations clustered in a logical or intuitive manner, not displaying more than what is necessary at the moment of interaction with a particular panel setting the context.

In the preferred embodiment, the remote according to invention allows its users to create an arbitrary number of macros with each up to 255 steps. Each step or action can be an IR (or RF) code, a 0.1 to 99.9 second delay, a reference to any other button, key or apparatus listed on the remote's GUI or a jump to any panel of the GUI. The user can create a macro by pushing a sequence of buttons and keys. The device records the sequence in the order of the steps entered. Afterwards, the user can view the macro as a scrollable list of steps, add more steps, select and delete steps, select and move steps up or down the macro, and select and adjust the value of delay steps. The visual feedback and editing capabilities are believed to be unique for an AV remote control. The remote for a home theater is thus provided with a macro creation/editing mode that has authoring tools accessible via the remote's GUI. One of the editing tools lets the user move a selected macro step visibly up or down the list of steps on the GUI. This authoring tool in the remote of the invention greatly improves the ergonomic character and user-friendliness and is believed to eventually support wider acceptance of home theater equipment.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is explained by way of example and with reference to the accompanying drawings, wherein:

FIG. **1** is a block diagram of a home theater system;

FIG. **2** is a diagram of a remote control device; and

FIGS. **3–6** are diagrams of the GUI of the remote when the user edits a macro.

Throughout the figures, same reference numerals indicate similar or corresponding features.

PREFERRED EMBODIMENTS

FIG. **1** is a block diagram of a home theater system **100**. The system comprises, for example, the following pieces of consumer equipment: an audio amplifier **102**; a cable-TV box **104**; a CD/DVD player **106**; a TV tuner **108**; a VCR **110**; a home security system **112**, a lighting control system **114** for control of luminaires (not shown) and a remote control device **116**. Equipment **102–114** is capable of being controlled by, in this example, IR commands (alternatively or subsidiarily RF commands). Remote **116** is capable of controlling it all through issuing the proper IR commands. Remote **116** has also the capability of being programmed for a macro instruction that is being executed upon a single user-interaction with remote **116**. For example, the user can

program a macro that first turns on home security system **112**, then audio processor **102**, after this turns on CD player **106**, further shows on remote **116** the panel for adjusting the volume of the music, thereafter selects a particular piece of music based on the content available from a CD carousel (not shown), and shows on remote **116** the panel for setting the lighting of the room at a desired level and finally let the GUI of remote **116** return to a home panel. In the invention, remote **116** allows its users to program adjustable time delays in the macro, for example, a delay of 10 seconds for the user to select the desired sound volume in the above example. User-intervention during execution of the macro is an option. It may, however, be more convenient to have as a last step of the macro a jump to a specific control panel, or simply the home panel of remote control device **116**.

Remote control device **116** has a GUI **118** on a display, e.g., an LCD, that has a touch screen functionality, e.g., a resistive tablet. Remote **116** further has hard keys **120** and **122**. The GUI has user-selectable menu options that are organized in various panels and sections of panels. A panel is, for example, a cluster of soft keys displayed together. The major user-selectable options for control of system **100** are provided as soft-keys in GUI **118**. Hard keys **120** and **122** are programmed so as to provide quick user-control dependent on the operational mode of remote **116**. For example, left-hand key **120** is mapped onto the "Stop" function when remote **116** is set into any mode for control of a transport functionality of equipment that is supplying video and/or audio output, e.g., VCR **110** or DVD player **106**, and right-hand key **122** is mapped onto the corresponding "Play" function.

FIG. **2** is a block diagram of remote control device **116** for control of home theater equipment **201** through a wireless protocol in IR or RF. Remote **116** comprises a display **202** for display of GUI **118** and a touch screen **204** for activating a user-selectable option of GUI **118**. Display **202** comprises an LCD. Touch screen **204** is, for example, a resistive tablet. For more background on such input devices, see, for example, U.S. Pat. Nos. 5,402,151; 5,231,381; 5,777,607 and 5,767,458 of Philips Electronics, all incorporated herein by reference. In the preferred embodiment, LCD **202** and touch screen **204** are physically integrated, and the combination has screen dimensions of about 77×58 mm for the active area, and about 81×62 mm for the visible area. The screen resolution is about 4 pixels per mm. Remote **116** comprises a frame buffer **206**, an off-screen memory **208**, a microprocessor **210**, an object store **212** and a non-volatile memory **214**. Frame buffer **206** is coupled to display **202** and stores the information content shown on display **202**. Off-screen buffer **208** stores bitmaps that are mapped into frame buffer **206** under control of a software bitblitter run on microprocessor **210**. Microprocessor **210** receives user-input via touch screen **204** and translates the input into associated GUI actions via framebuffer **206** and off-screen buffer **208**. Object store **212** lists the information to be rendered in off-screen buffer **208** and/or in frame buffer **206** by microprocessor **210**. The information is stored in object store **212** as objects. Non-volatile memory **214** stores data files that describe the macro's and the individual macro steps. This description is in a universal format, e.g., <"type of action"; "parameters">, so that it can be interpreted platform-independently. Interpretation creates objects, e.g., in C++, for object store **212**. In the preferred embodiment, processor **210** is a MC68328 of Motorola. Buffers **206** and **208**, together with object store **212** are implemented in an SRAM **216** such as a 512 kbyte SRAM, e.g., a KM616Y4000BLT-7L of Samsung. Non-volatile memory **214** is, for example, a 1 Mbyte flash memory T E28F800B3B90 of Intel.

FIGS. **3–6** illustrate the GUI of remote **116** in the macro editing mode for recording on the VCR a TV program. The GUI shows a portion of a recorded macro with the following steps: "TV-on" (turn on the TV); "TV-3" (select channel 3); "Delay—0.5 sec" (wait for 0.5 seconds before issuing the next command); "VCR-on" (turn the VCR on); and "VCR-TV/VCR" (use the TV as source for the VCR). The GUI provides tools for editing the macro such as a scroll key **302**, a delay key **304**, arrow keys **306** and **308**, and a delete key **310**. Tapping touch screen **204** at the location of a specific one of the steps shown selects this step for further operations. Scroll key **302** lets the user scroll the macro to bring into view further steps of the macro under consideration. Touching delay key **304** insert a delay into the macro. The delay step is added, for example, at the end of the macro. Arrow keys **306** and **308** move a selected step down or up the sequence of the macro. Key **310** deletes a selected step. In order to add a new command, the user has to go to an other panel of the GUI, e.g., the apparatus controls (not shown here) and enter the command with remote **116** in the macro creation mode. The new command is added to the sequence, e.g., at the end, and can be repositioned by selecting and arrow keys **306** and **308**. Other procedures can be implemented to insert the new command at the proper position. The example of FIGS. **3–6** illustrates how a macro is edited that has already been programmed. In the example, the step "Delay—0.5 sec" in FIG. **3** is selected by tapping on it. The GUI in FIG. **4** shows that the selected step is highlighted. In addition, delay control keys **402** and **404** pop up. Keys **402** and **404** are used to increase or decrease the length of the delay. The GUI in FIG. **5** shows the selected delay step moved up one step the sequence by use of arrow key **308**. The GUI in FIG. **6** shows the re-positioned delay step, now with an increased delay of 5.0 sec. instead of 0.5 sec. through use of key **402**. Tapping a "CLOSE" key **602** closes the macro and causes the GUI to return to the use-mode after writing the macro to flash memory **214**.

When the user puts remote **116** into the macro editing mode and selects a particular macro, the corresponding data are retrieved from flash memory **214**, interpreted by processor **210**, the resulting objects stored in SRAM **216**, and converted to one or more bitmaps that get displayed on display **202**. If the editing of the macro does not add more steps, the data for the edited macro is saved and stored at the addresses in flash memory **214** that held the previous version. If the editing leads to a longer sequence, the macro data is stored at new addresses in flash memory, and reference to the previous addresses is rendered invalid. Garbage collection can be used to delete data that are not used.

I claim:

1. A remote control device for remote control of equipment, the device having a display for display of a GUI with a plurality of user-selectable options, wherein:

the device enables the user to program a macro comprising a sequence of multiple ones of the control options;

the GUI enables the user to view the sequence as a scrollable list of steps;

the GUI enables the user to edit the list by selecting at least one specific one of the steps at a first location in the list, moving the specific step along the sequence, and inserting the specific step into a second location of the list.

2. The device of claim 1, wherein:

the control options are organized in two or more panels for individual display on the GUI; and

the device enables the user to program the macro comprising a particular step that causes the GUI to jump to a specific one of the panels.

3. The device of claim 1, wherein the device has a touch screen for enabling user-selection of the options.

4. The device of claim 1, wherein the GUI changes its appearance through animation upon selection of a particular one of the user-selectable options.

5. The device of claim 1, wherein the macro is programmable to be activated under a softkey on the GUI.

* * * * *

US005926178A

United States Patent [19]

Kurtenbach

[11] Patent Number: 5,926,178

[45] Date of Patent: Jul. 20, 1999

[54] **DISPLAY AND CONTROL OF MENUS WITH RADIAL AND LINEAR PORTIONS**

[75] Inventor: **Gordon P. Kurtenbach**, Toronto, Canada

[73] Assignee: **Silicon Graphics, Inc.**, Mountain View, Calif.

[21] Appl. No.: **08/840,725**

[22] Filed: **Apr. 1, 1997**

Related U.S. Application Data

[63] Continuation of application No. 08/469,913, Jun. 6, 1995., Pat. No. 5,689,667

[51] **Int. Cl.**[6] **G06F 3/00**

[52] **U.S. Cl.** **345/352**; 345/353; 345/146

[58] **Field of Search** 345/326–358, 345/146, 145

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,303,388	4/1994	Kreitman et al.	345/349
5,500,935	3/1996	Moran et al.	345/352
5,524,196	6/1996	Blades	345/353
5,559,944	9/1996	Ono	345/353
5,596,699	1/1997	Driskell	345/352
5,617,114	4/1997	Bier et al.	345/146 X
5,701,424	12/1997	Atkinson	345/353
5,721,853	2/1998	Smith	345/353

OTHER PUBLICATIONS

Foley et al., "Computer Graphics: Principles and Practice", Addison–Wesley, pp. 364–367, Nov. 1993.

Gordon P. Kurtenbach, "An Empirical Evaluation of Some Articulatory and Cognitive Aspects of Marking Menus", *Human–Computer Interaction*, 1993, vol. 8, pp. 1–23.

Gordon Kurtenbach and William Buxton, "The Limits Of Expert Performance Using Hierarchic Marking Menus", INTERCHI '93, Apr. 24–29, 1993, pp. 482–487.

G. Kurtenbach, T.P. Moran and W. Buxton, "Contextual Animation of Gestural Commands", Computer Graphics forum, vol. 13, (1994) No. 5, pp. 305–314.

Don Hopkins, "The Design and Implementation of Pie Menus", Dr. Dobb's Journal, Dec. 1991, pp. 16, 18, 22, 26.

Gordon Paul Kurtenbach, "The Design and Evaluation of Marking Menus", 1993, pp. 1–173.

Gordon Kurtenbach and William Buxton, "User Learning and Performance with Marking Menus", Human Factors in Computing Systems CHI '94, Apr. 24–28, 1994, pp. 258–264.

LaLonde et al, "Pie Menus", JOOP, V. 2, N. 1, May/Jun. 1989, pp. 47–54.

Callahan et al., "An Empirical Comparison of Pie v. Linear Menus", ACM, 1988, pp. 95–100, (with illustrations).

Hopkins et al., "Pies Implementation, Evaluation, and Application of Circular Menus", UMCS Dept. tech. report, 1988, pp. 1–25.

Primary Examiner—John E. Breene

[57] **ABSTRACT**

A menu may be displayed combining a radial marking menu portion and a linear menu portion. Item selection in the linear portion is performed by location selection using a pointing device. Item selection in the marker portion is determined by the pattern of a stroke made by the pointing device with the system ignoring linear menu items across which the stoke completely passes. Linear menu items may be non-hierarchically related to the radial items. Linear and radial portions may be simultaneously produced on a display based on a single input event.

2 Claims, 9 Drawing Sheets

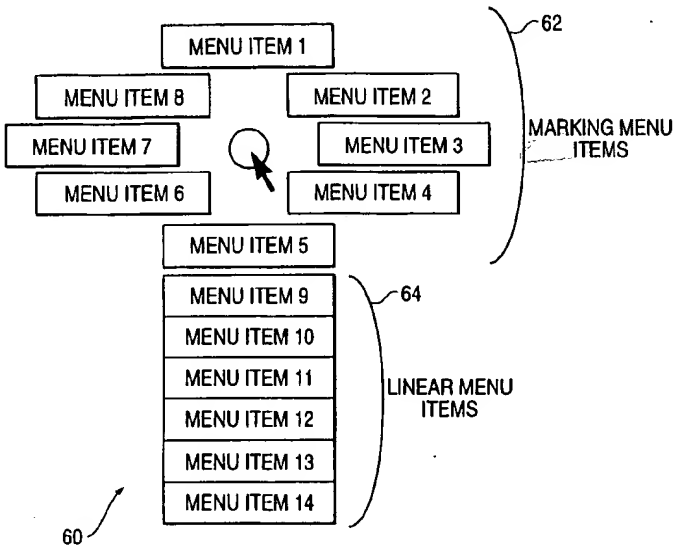


FIG. 1

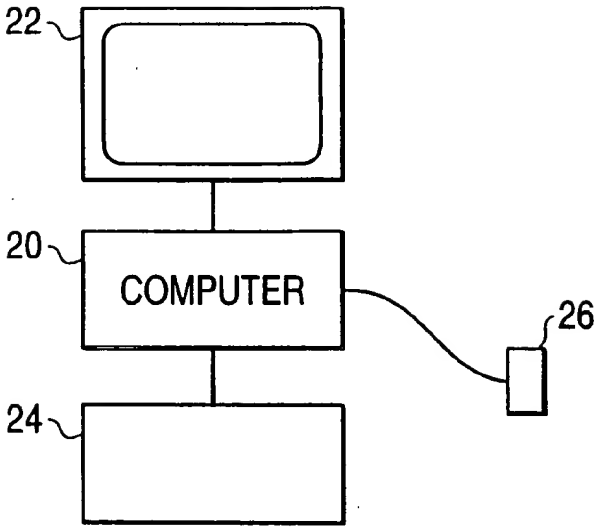

22
20
COMPUTER
26
24

FIG. 3

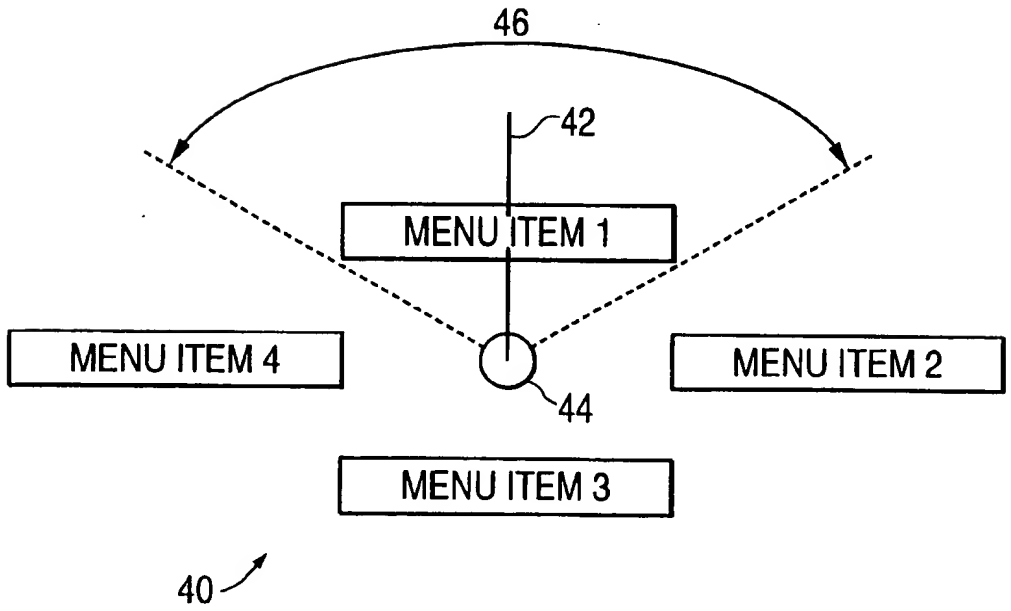

46
42
MENU ITEM 1
MENU ITEM 4
MENU ITEM 2
44
MENU ITEM 3
40

FIG. 2

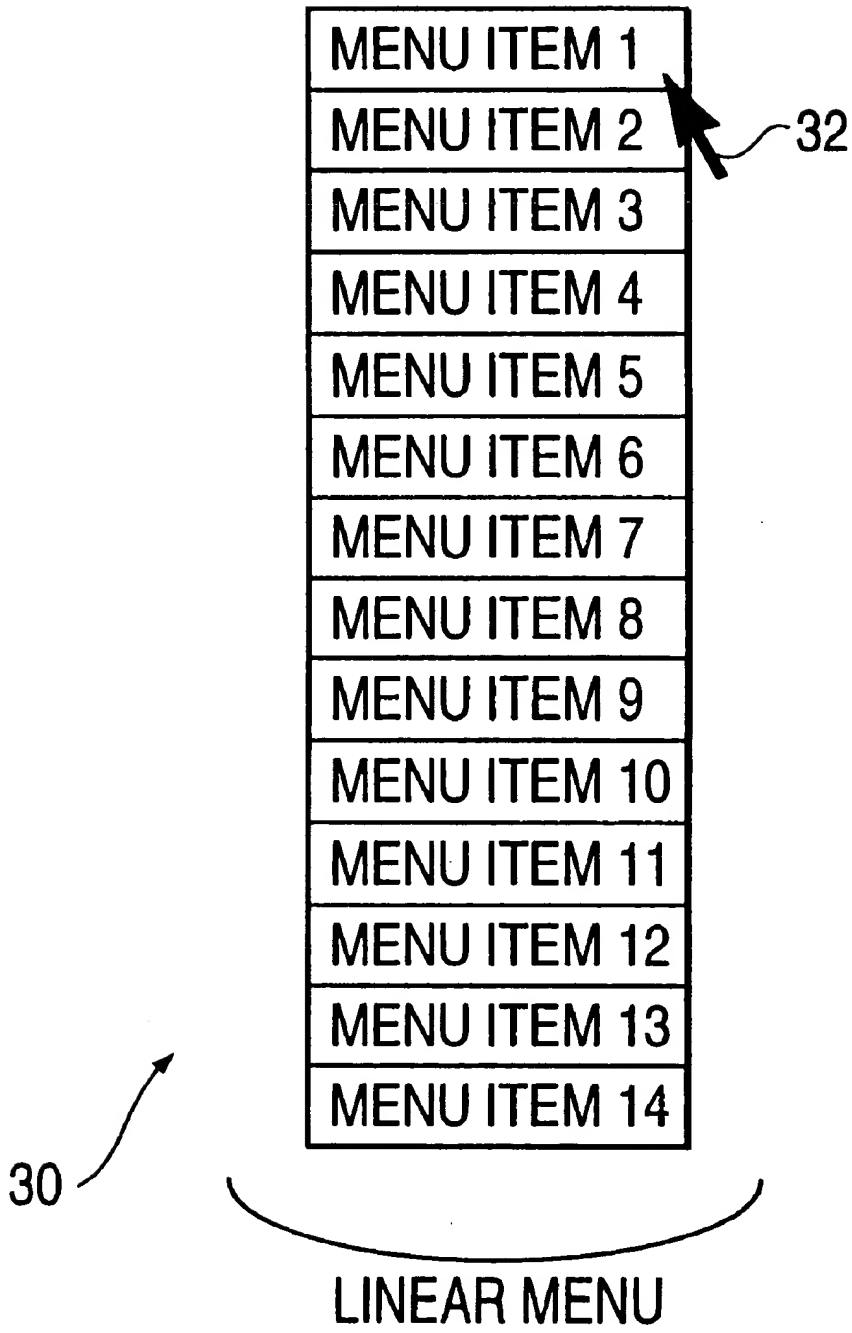

MENU ITEM 1
MENU ITEM 2
MENU ITEM 3
MENU ITEM 4
MENU ITEM 5
MENU ITEM 6
MENU ITEM 7
MENU ITEM 8
MENU ITEM 9
MENU ITEM 10
MENU ITEM 11
MENU ITEM 12
MENU ITEM 13
MENU ITEM 14
32
30
LINEAR MENU

FIG. 4

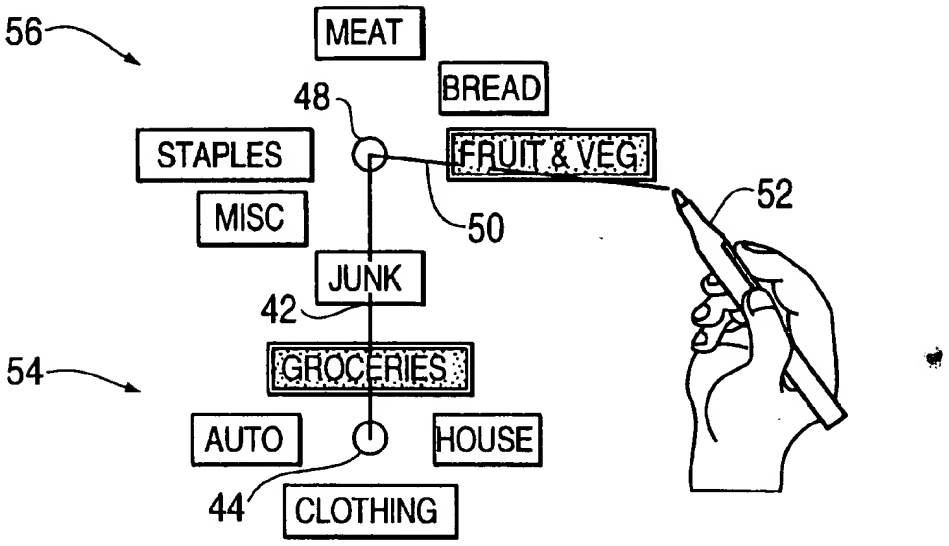

56
MEAT
BREAD
48
STAPLES
FRUIT & VEG
52
MISC
50
JUNK
42
54
GROCERIES
AUTO
HOUSE
44
CLOTHING

FIG. 5

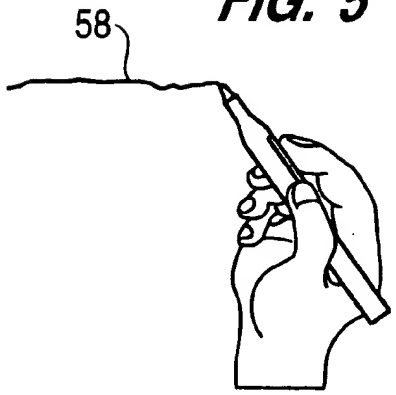
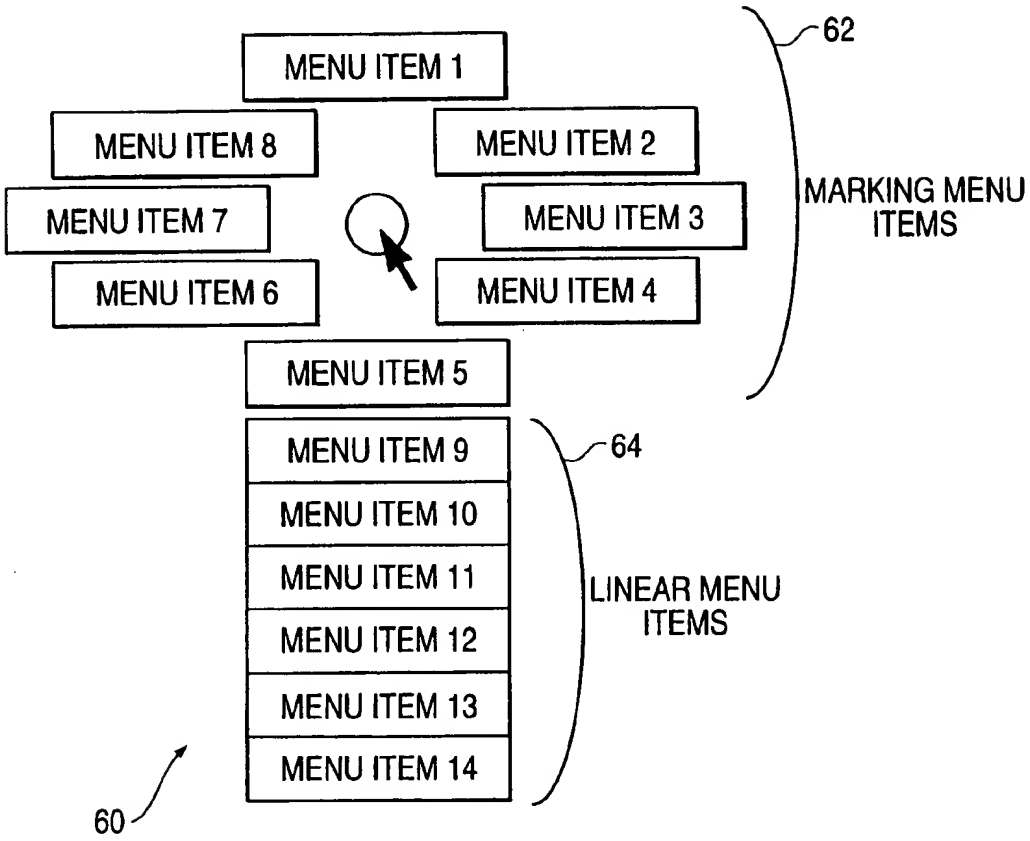

58

FIG. 6


MENU ITEM 1
MENU ITEM 8
MENU ITEM 2
MENU ITEM 7
MENU ITEM 3
MENU ITEM 6
MENU ITEM 4
MENU ITEM 5
62
MARKING MENU ITEMS
MENU ITEM 9
MENU ITEM 10
MENU ITEM 11
MENU ITEM 12
MENU ITEM 13
MENU ITEM 14
64
LINEAR MENU ITEMS
60

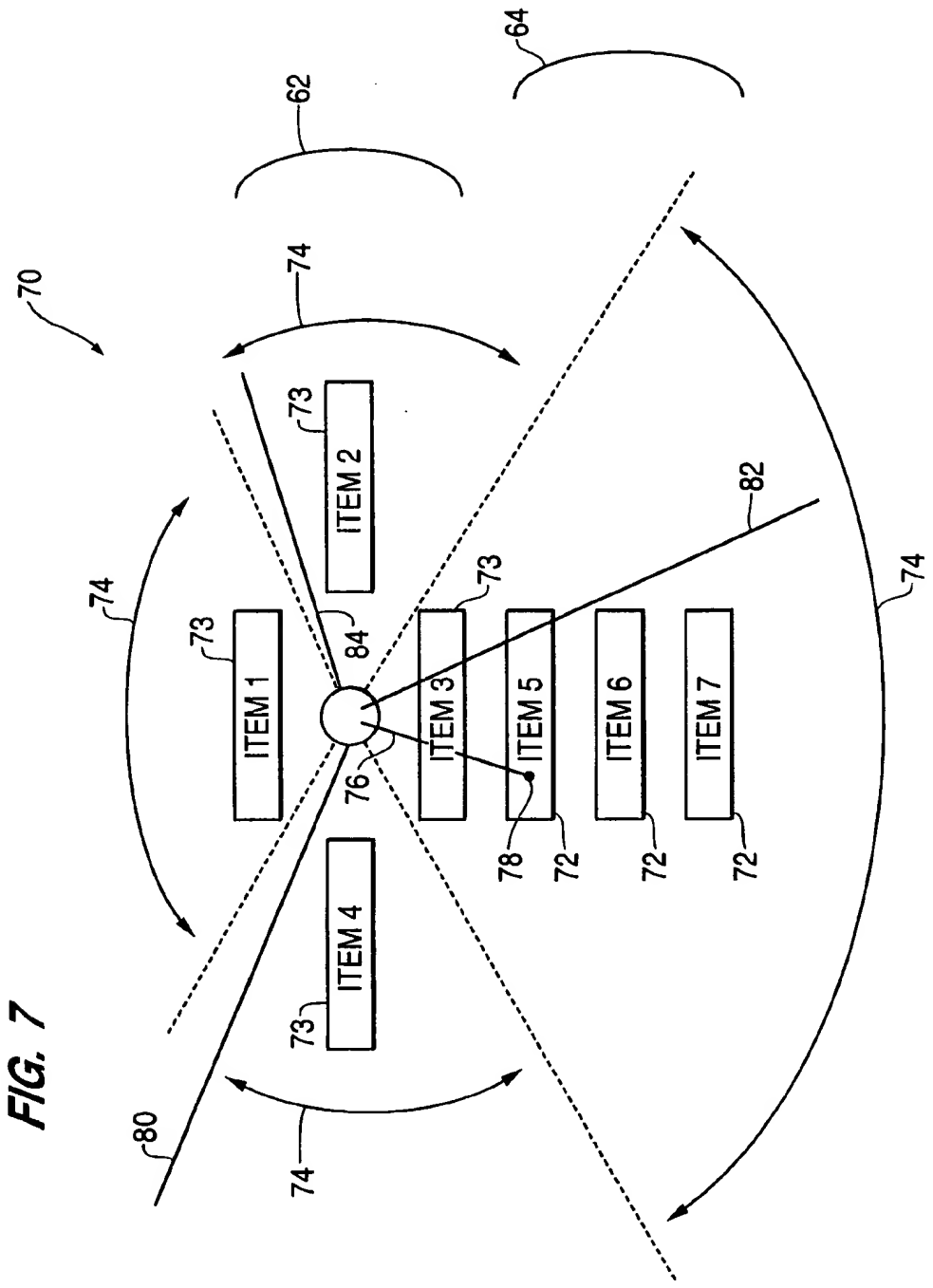

64
62
70
74
73
ITEM 2
82
74
73
ITEM 1
84
73
ITEM 3
ITEM 5
ITEM 6
ITEM 7
74
76
78
72
72
72
ITEM 4
73
80
74

FIG. 7

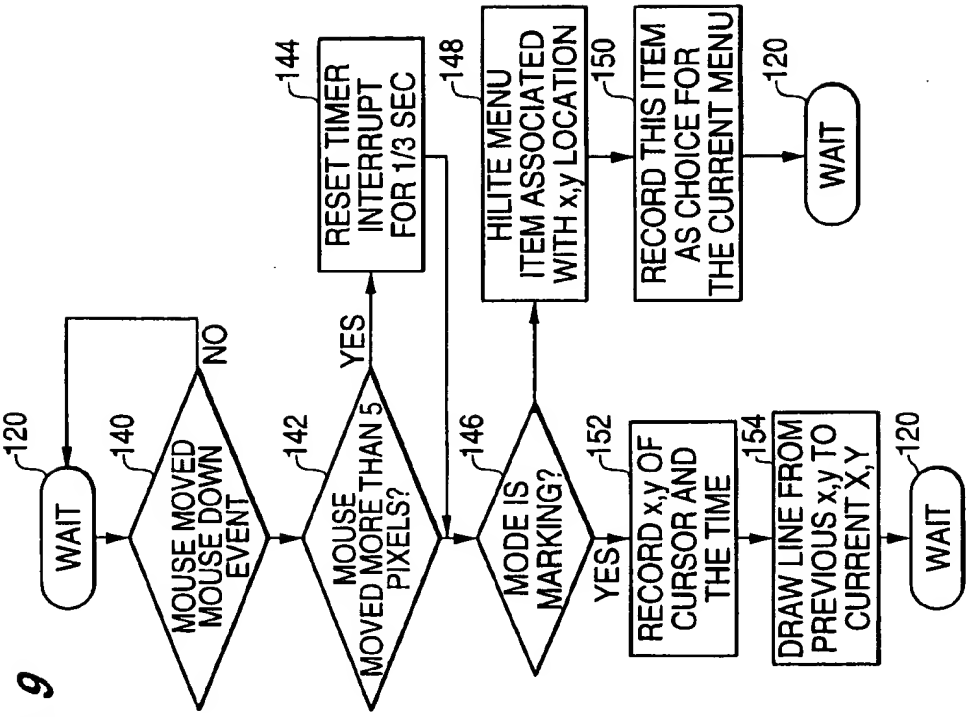

120
WAIT
140
MOUSE MOVED MOUSE DOWN EVENT
NO
142
MOUSE MOVED MORE THAN 5 PIXELS?
YES
144
RESET TIMER INTERRUPT FOR 1/3 SEC
146
MODE IS MARKING?
YES
148
HILITE MENU ITEM ASSOCIATED WITH x,y LOCATION
150
RECORD THIS ITEM AS CHOICE FOR THE CURRENT MENU
120
WAIT
152
RECORD x,y OF CURSOR AND THE TIME
154
DRAW LINE FROM PREVIOUS x,y TO CURRENT X,Y
120
WAIT

FIG. 9

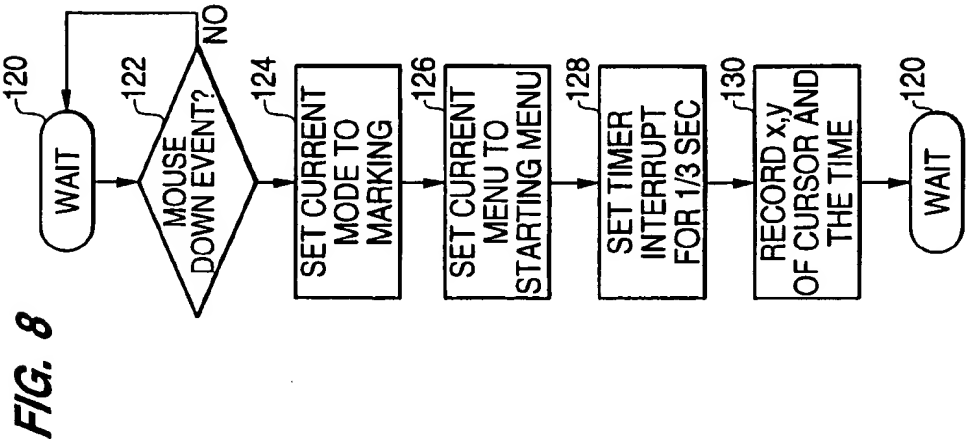

120
WAIT
122
MOUSE DOWN EVENT?
NO
124
SET CURRENT MODE TO MARKING
126
SET CURRENT MENU TO STARTING MENU
128
SET TIMER INTERRUPT FOR 1/3 SEC
130
RECORD x,y OF CURSOR AND THE TIME
120
WAIT

FIG. 8

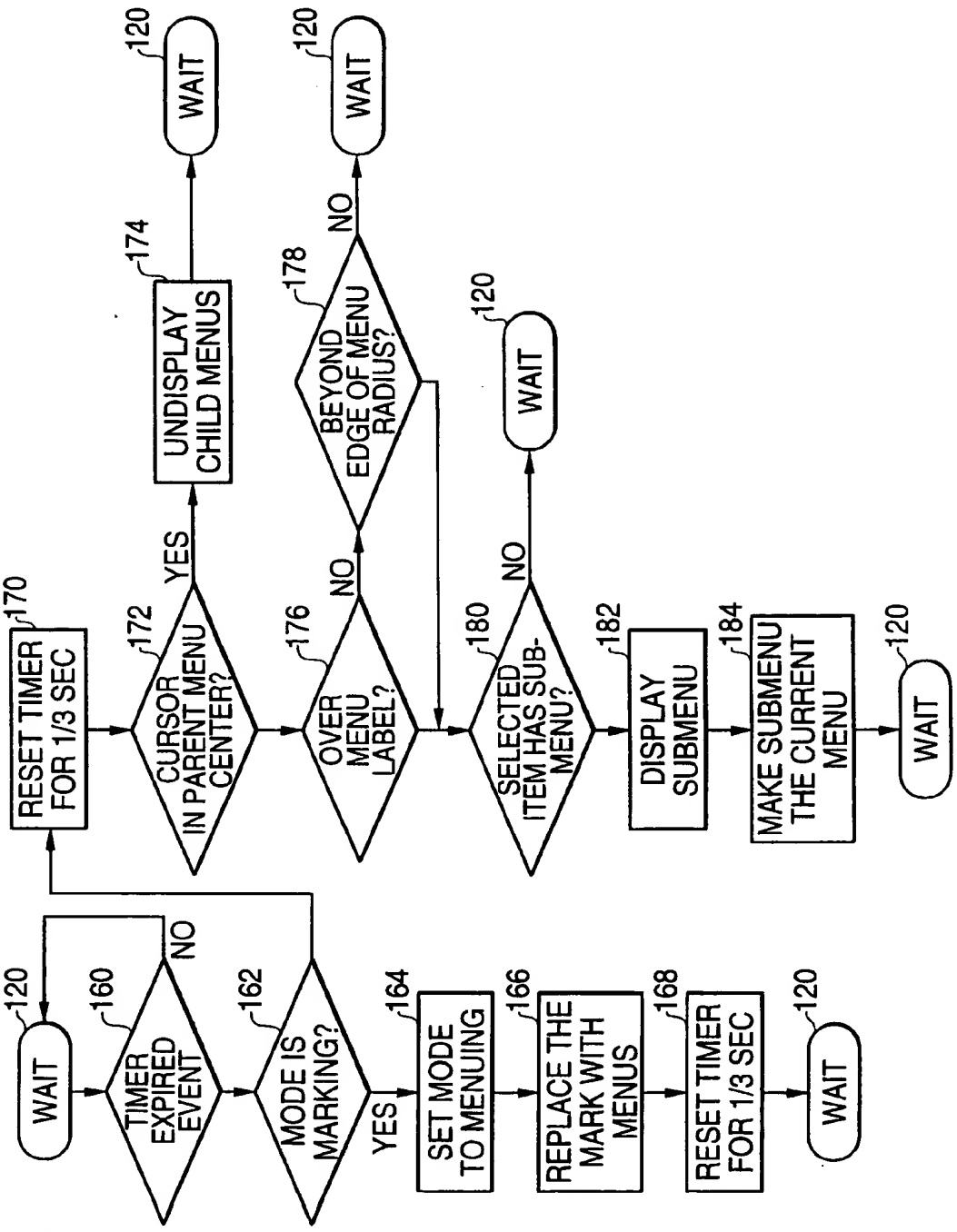

120
WAIT
160
TIMER EXPIRED EVENT
NO
162
MODE IS MARKING?
YES
164
SET MODE TO MENUING
166
REPLACE THE MARK WITH MENUS
168
RESET TIMER FOR 1/3 SEC
120
WAIT
170
RESET TIMER FOR 1/3 SEC
172
CURSOR IN PARENT MENU CENTER?
YES
174
UNDISPLAY CHILD MENUS
120
WAIT
176
OVER MENU LABEL?
NO
178
BEYOND EDGE OF MENU RADIUS?
NO
120
WAIT
180
SELECTED ITEM HAS SUB-MENU?
NO
120
WAIT
182
DISPLAY SUBMENU
184
MAKE SUBMENU THE CURRENT MENU
120
WAIT

FIG. 10

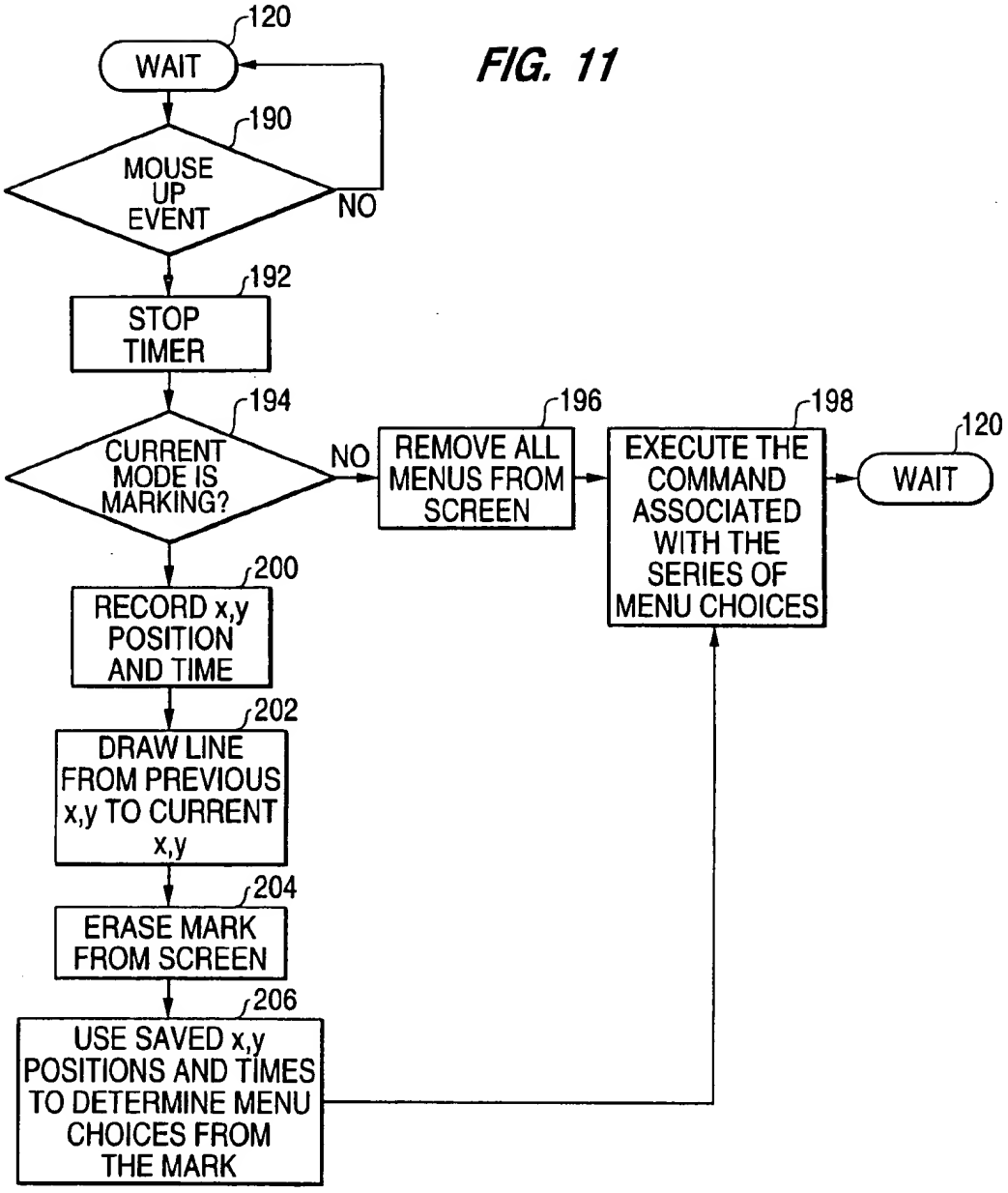

120
WAIT
190
MOUSE UP EVENT
NO
192
STOP TIMER
194
CURRENT MODE IS MARKING?
NO
196
REMOVE ALL MENUS FROM SCREEN
198
EXECUTE THE COMMAND ASSOCIATED WITH THE SERIES OF MENU CHOICES
120
WAIT
200
RECORD x,y POSITION AND TIME
202
DRAW LINE FROM PREVIOUS x,y TO CURRENT x,y
204
ERASE MARK FROM SCREEN
206
USE SAVED x,y POSITIONS AND TIMES TO DETERMINE MENU CHOICES FROM THE MARK

FIG. 11

FIG. 12

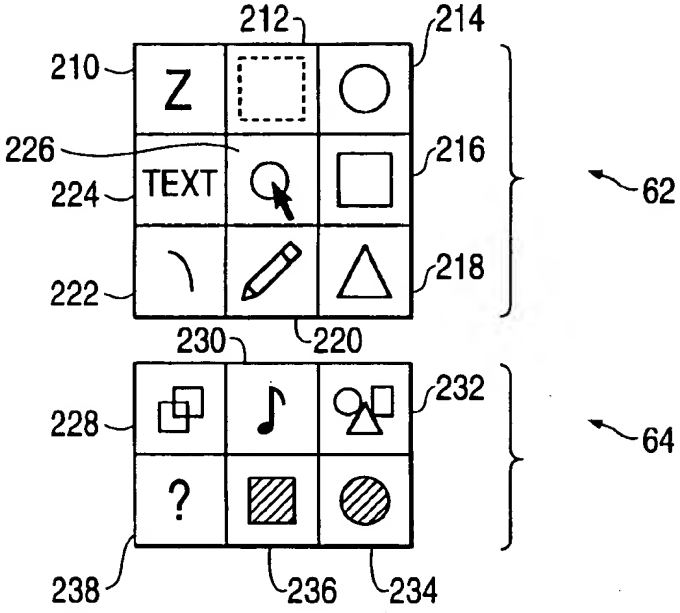
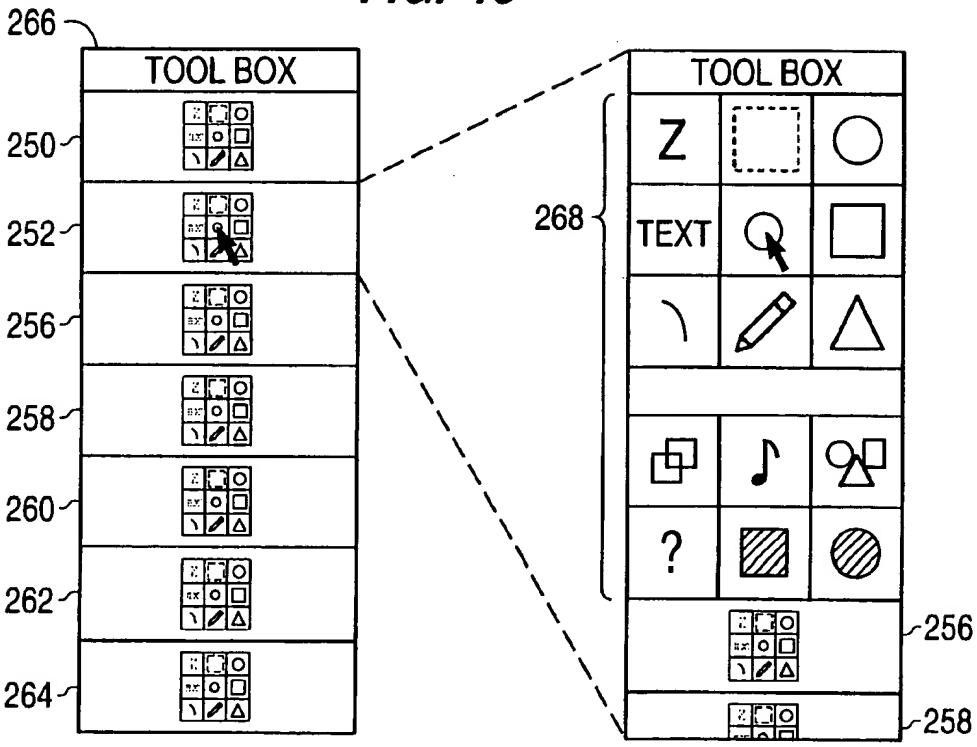

212
214
210
Z
226
TEXT
224
216
62
222
218
230
220
228
232
64
?
238
236
234

FIG. 13


266
TOOL BOX
250
252
256
258
260
262
264
TOOL BOX
Z
268
TEXT
?
256
258

DISPLAY AND CONTROL OF MENUS WITH RADIAL AND LINEAR PORTIONS

This application is a continuation of application Ser. No. 08/469,913, filed Jun. 6, 1995, now U.S. Pat. No. 5,689,667.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention is directed to a method and apparatus for producing, controlling and displaying menus that combine radial marking menus and linear menus and, more particularly, to a system that combines a pattern selectable menu with a location selectable menu into a single display.

2. Description of the Related Art

Menus selection in modern user interfaces is a fundamental and frequently used operation. In general, the length of time it takes to select from a menu is small (seconds or fractions of second). However, because menu selection is a high frequency operation in some types of applications, reducing the time involved in selecting from a menu can result in significant time savings in the overall time it takes to accomplish a larger task. It is for this very reason that menu selection speed up ("accelerator") techniques like "hot keys" and marking menus are popular.

Each accelerator technique has its own advantages and disadvantages. For example, hot keys can select from a large set of menu items, for example, every key on the keyboard can be used as hot key. However, this is at the cost of the user remembering the associations between menu items and keys. Hot keys are also not effective if a user is working on a system that does not include a keyboard or the user must move between the keyboard and another device while operating the interface. For example, the user may have to move the same hand from a mouse, to the keyboard, and then back to the mouse.

Marking menus is a pop up menu technology that displays menu items in a circle, around the cursor whereas traditional menus (linear menus) display menu items in a linear fashion, for example, from top to bottom. Because of this difference, item selection can be performed more rapidly with marking menus than with linear menus. For example, a user can select from marking menus by "flicking" or making a stroke of the cursor in the direction of a desired menu item. Thus, the user does not have to wait for the menu to be displayed to select from it. However, as the number of items in a marking menu increases, rapid selection without popping up the menu becomes difficult because the angular difference between menu items becomes small and difficult for the user to articulate. To combat this problem, the number of items in a marking menus is generally limited to eight or less.

What is needed is a combination of a radial marking menu and a linear menu in the same display such that the number of items in the menu can be increased beyond eight items while still permitting rapid selection for the items of the marking menu using a marking or stroke pattern and selection of the items of the linear menu using a locational method.

SUMMARY OF THE INVENTION

It is an object of the present invention to combine radial marking menus with linear menus in the same display.

It is another object of the present invention to combine marking menu selection techniques with location selection techniques.

It is an additional object of the present invention to provide a menu display in which selection of some items are accelerated.

It is also an object of the present invention to provide a process that distinguishes between linear (or location) based menu selection and pattern-based menu selection.

The above objects can be attained by a system that combines a radial marking menu, in which item selection is performed by marking a menu item with a cursor mark, with a linear menu, in which item selection is performed by selection at a particular location. The system distinguishes between parts of the display in which the linear menu is located and parts of the display in which the radial marking menu is located.

These together with other objects and advantages which will be subsequently apparent, reside in the details of construction and operation as more fully hereinafter described and claimed, reference being had to the accompanying drawings forming a part hereof, wherein like numerals refer to like parts throughout.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** illustrates typical hardware of the present invention;

FIG. **2** illustrates a linear or location selection type menu;

FIG. **3** illustrates a radial marking type menu;

FIG. **4** illustrates submenu selection using a marking menu;

FIG. **5** depicts selection using a marking pattern without producing a display;

FIG. **6** shows a combined radial marker and linear menu according to the present invention;

FIG. **7** illustrates selection operations of the invention;

FIGS. **8–11** comprise flowcharts of the steps of the present invention; and

FIGS. **12** and **13** depict an icon menus and expansion thereof from a tool pallet.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention typically has a hardware configuration, as illustrated in FIG. **1**, that includes a computer **20** which generates and displays menus on a display **22**. A user interacts with the display **22** through the computer system **20** using an interface device, such as a keyboard **24**, and a pointing device **26**, such as a mouse. The computer **20** includes the conventional hardware necessary to interact with a pointing device **26**, such as motion detection circuits and button depression detection circuits, and the appropriate storage media, such as a hard disk or a floppy disk on which the process discussed herein is stored. The computer **20** can be any type of computer from a personal/portable computer to a main frame computer with the typical computer being a workstation type computer that is used for high volume production operations in which menu selection is a significant portion of the operations performed by a user, such as in CAD/CAM and animation operations. The display **22** can also be any type of display that is capable of displaying a menu, such as a CRT display or a liquid crystal display. The pointing device **26** can be any type of pointing device including a pen, a mouse, a track ball, a track point or a joy stick, either separate from or part of the keyboard **24**. The display **22** and pointing device **26** can also be combined into a single device, such as touch sensitive screen.

The present invention combines a radial marking menu with a linear menu into a single display where the radial and linear portions are associated with each other and displayed

simultaneously. The present invention allows the menus to be selected using the pattern and location techniques of these two types of menus. The selection and display arrangement of linear and radial menus will be discussed before discussing their combination.

A linear menu **30**, as illustrated in FIG. **2**, is typically a linear array of menu items where the number shown in FIG. **2** is fourteen items (**1–14**) but where the array shown is a single dimensional array. The number of items can, of course, be less than or greater than 14 and the items can be arranged in two or more dimensions or even in an arbitrary arrangement. The main feature of a linear menu is the way in which it is used to select a desired item. Typically, a pointer (or cursor) **32**, such as an arrow as shown, is moved to the location or area defined by the menu item desired to be selected by manipulating the pointing device **26** and the selection is made by depressing (or releasing) a button that is typically associated with the pointing device **26** but which could be in a separate interface device, such as the keyboard **24**. The computer **10** determines the location of the pointer **32** at the time the button is depressed (or released) and if the pointer is within the area defined by a menu item, that item is selected. If the pointer is not within any areas defined by menu items nothing is selected. That is, linear menus use the location of the pointer **32** to determine the item selected. If the user depresses (or releases) the selection button while the pointer **32** is positioned over Menu Item **1**, as shown in FIG. **2**, Menu Item **1** would be selected. Using a linear menu requires that the user position the pointer **32** in the area of the display defined by the item to be selected. This can take a considerable amount of time for the user and accurate positioning is required. The accurate positioning is even more critical when the item is an icon that can be rather small.

Radial menus include two types: pie menus and marking menus. Pie menus are typically used in item selection using the location principles of linear menus as discussed above. Marking menus operate on the principle of the direction of cursor or pointer motion as being the basis for item selection. Marking is a menu selection operation which works as follows. A user activates the selection indicator of the pointing device **26**, such as by pressing down on the screen with a pen or holding down the button on a mouse and waits for a short interval of time (approximately ⅓ second). A radial menu **40**, such as illustrated in FIG. **3**, then appears ("pops-up") directly under the tip of the pen **52**. A user then highlights an item by keeping the pen pressed and making a stroke **42** towards the desired item. The stroke **42** is created by the computer constantly redrawing a straight line between the center **44** of the menu **40** and the current location of the pointer (that is, the stroke **42** behaves like a "rubber band line" between the center of the menu and the pointer). The computer **20** highlights a particular item when the pointer moves into the angular range associated with that menu item (for example, angle range **46**, associated with Menu Item **1** in FIG. **3**). If the item has no sub-menu, the highlighted item can be selected by lifting the pen. If the item does have a sub-menu, and the user stops moving the pointer, the submenu is displayed with the center **48** of the new menu under the pen **52** (see FIG. **4**). The user then continues, from the new center **48**, selecting the newly displayed sub-menu by providing another stroke **50** with the pen **52**. Lifting the pen **52** will cause the current series of highlighted items to be selected. In the example of FIG. **4** the items selected are "Groceries" in the main menu **54** and "Fruit & Veg" in the submenu **56**. The menus are then removed from the screen. At any time a user can indicate "no selection" of the submenu **56** by moving the pen back to the center (**48**) of the submenu before lifting, or change the selection by moving the pen to highlight another item before lifting. Finally, a user can "back-up" to a previous menu by pointing to its center **44**.

Another, faster, way to make a selection without popping up the menu is by drawing a mark **58** as illustrated in FIG. **5**. A mark can be drawn by pressing the pen down and immediately moving. The direction of the mark **58** dictates the particular item selected from the menu. In the present invention if only a mark is made, as will be discussed in more detail later, and the menu is not allowed to pop-up, the system interprets the mark as if the linear portion of the present invention does not exist. This is called the "marking" mode as opposed to the "menuing" mode when the combined marking and linear menu is displayed which will be discussed in more detail later.

Thus, marking menus allow items to be selected in two different ways. Using the method of FIG. **4** radial menus can be sequentially displayed and selections made. The method of FIG. **5** uses marking without menu display to make the same selection. The first method is good when the user is unfamiliar with the menu. The second method is good when the user is familiar with the menu and wants to avoid waiting for the display of the menu.

Marking menus avoid some of the problems of hot keys. A user does not have to remember the association between keys and menu items. The user only needs to remember the spatial layout of menu items. Typically users very quickly learn the spatial location of menu items, especially for frequently used commands. Marking menus also do not require a keyboard for operation. However, because human capability is limited in the accuracy in which angles can drawn, the number of menu items in marking menus is generally limited to eight items.

In many situations modern user interfaces use menus with more than eight items. For example, the number of items in pop-up menus in the ALIAS V6 package ranges from 8 to 15 items, with an average of 12.75 items per pop-menu. Higher number of items can be used in marking menus by making the menus hierarchic. In this case rapid menu selection requires a zig-zag type line with pauses rather than just a simple stroke. For example, a 64 item menu can be made up of an eight item menu where each item leads to another eight item submenu. Thus, if a user interface designer wishes to convert an application which uses linear menus into one which uses marking menus, the designer must convert any menu that takes more than eight items into an hierarchic menu. This conversion process requires extra work and linear menu items may not decompose naturally into hierarchic categories.

The present invention combines marking menus and linear menus into a single menu **60**, as illustrated in FIG. **6**, such that converting linear menus into a marking menu does not require the decomposition of menu items into categories. In the invention up to the first eight items of the traditional linear menu that are the most highly selected are mapped directly to a marking menu portion **62**. The additional ("overflow") items are mapped to a linear menu portion **64**, which is displayed at the same time the marking menu portion **62** is displayed. The linear portion **64** is shown below the marker portion **62**, however, the linear portion position can be varied as desired as long as it appears simultaneously on the same display.

The present invention, as depicted in FIG. **6**, has the following advantages. First, as stated before, conversion of

linear menus to this format is simple. No decomposition into hierarchic categories is needed and more than eight item per menu can be used. Second, accelerated selection can be performed on the first eight items, such that time savings can be realized over traditional linear menus. Third, because some of the items in the menu **60** are laid out side by side as opposed to top to bottom, the overall height of menu **60** is reduced. Finally, this type of menu **60** takes up approximately the same screen space as a tradition linear menu and therefore this new menu **60** can be displayed at the same location as tradition linear menus (for example, the menu can be pop-up or pop-down, etc.).

The invention has some characteristics different from traditional linear menus. First, the user must pause to display the menu before selecting an item from the linear portion **64** of the menu. This is not a great disadvantage since these items are by definition supposed to be infrequently used commands. If the position where the menu pops-up is very close the edge of screen there may not be enough room to select some of the items from the radial portion **62** of menu using a flick. For example, suppose the menu **60** pops up when a user presses the mouse button down right in the top left corner of the screen. At this point there is no room to "flick" the cursor up or to the left to select some of the menu items. In practice, this is not a serious problem since there are very few situations where a menu is popped up right along the edge for the screen. Also, the amount of room needed for a flick can be very small. Finally, the radial portion **62** of menu **60** represents a grouping of menu items that is based on frequency of selection. In linear menus, items are generally grouped together based on function. Therefore, the conversion to this new menu system might require rearrangement of items in a menu. Fortunately, the particular grouping of menu items has little value once a user becomes familiar with a menu layout and therefore this is not a serious problem in the long run.

The creation of such a combined menu requires that a displayed menu **70**, as illustrated in FIG. **7**, be divided into different regions. In the linear region which is defined by each of the areas of the individual menu items, "display buttons" or selection regions **72** are defined. In the marking region, which is all of the display outside the display buttons **72** and **73**, the selection regions are defined by angular ranges **74**. During operation if a stroke or mark, such as stroke **76**, has an end point **78** that is within a linear selection region **72** or within a label **73** (and the pen is lifted at that point), the item of the end point **78** is selected which in this example is Item **5**. If a stroke **80** is in a region of the marking menu portion, the item selected is the item of the coinciding range or wedge which in this particular example is Item **4**. If a stroke **82** ends in a marker region **74** after crossing one or more linear regions **72**, the item selected is the item of the coinciding wedge or marker region which in this particular example is Item **3**. That is, whenever the end point of the stroke does not lie within a linear region **72** (or label **73**), the coinciding or corresponding marker region item is selected. A radial menu label can fall outside an angular range associated with that label, however, pointing to a menu item overrides the angular ranges. For example, a portion of Menu Item **2** (the area of the label for Item **2**) in FIG. **6** can lie, depending on how the angular ranges of each item are defined, in the angular range for Menu Item **3**. However, pointing directly to the label of Menu Item **2** will highlight it even if the pointer is in the angular range of Menu Item **3**. Linear menu items, such as **72** in FIG. **7**, have no "angular range" features and the user must position the pointer directly over the label for the item to be selected.

The invention behaves differently from the description above when the user does not display the menus but draws a mark. In this case, the mark is interpreted as if the linear menu items do not exist. For example, if a mark like **76** were drawn this would not result in the selection of linear menu item **5**, but in radial item menu **3**. This has the advantage that a user can quickly select radial menu items with a quick mark (a "flick") without accidently selecting linear menu items.

The present invention is directed to the combination of radial menus and linear menus in the same display. If the invention were to follow the typical rules for menu selection in the prior art this would mean that all menu selections (either by displaying a menu or by drawing a mark) would be based on the location of the cursor at the end of the stroke. This in turn would result in selection errors when a user tries to select from the menu of the present invention quickly. Thus, the advantage and distinction of the present invention is apparent.

The operation of the process of determining the selection is illustrated in the flowcharts of FIGS. **8–11**. Typically such a process is interrupt driven with interrupts occurring based on an interrupt timer or based on events, such as the depression/release of a mouse button or the detection of mouse movement. As can be seen from a visual review of FIGS. **8–11** the operations performed begin with an exit from a wait state **120** and return to this state **120**. The exit from the wait state is caused by an event, which event is checked to determine the type of event as discussed in more detail below. The operations can generally be broken into pen-down—FIG. **8**, pen—drag FIG. **9**, pen-still—FIG. **11** and pen-up FIG. **11**.

The pen-down operation of FIG. **8** first checks **122** to determine whether the mouse button down event has occurred. If not, the system returns to the wait state **120**. Otherwise, the mode is set **124** to the marking menu mode and the menu selection is set **126** to the starting or highest menu in a hierarchy. A system timer is then set **128** to ensure that the system detects that the pen/mouse is no longer moving or is still. The current position of the cursor and the time are then recorded **130**.

In the pen-drag operation the first check performed, as depicted in FIG. **9**, is to determine **140** whether the event of the mouse moving while the button is down has occurred. If not, again the system returns to the wait state **120**. If so, a check **142** is made of the distance of movement of the cursor or pointer against a movement threshold. If the threshold has been exceeded the timer is reset **144**. In both cases the next step is a determination **146** concerning the mode. If the mode is not the marking mode, the item of the particular location of the cursor or pointer is highlighted and the item is recorded **150** as the current choice. If the mode is the marker mode, the current cursor location and the time is recorded.

In the pen-still operation (FIG. **10**), the first check **160** is to determine whether the timer event has occurred, if not, the wait state **120** is entered. In this operation the system also checks **162** on the mode. In the marker mode the mode is set **164** to the menu mode and the mark is replaced **166** by the appropriate menus as previously discussed. The time is then set **168** and the wait state is entered. When not in the marker mode, the system sets **170** the timer and determines **172** whether the cursor is in the center of the parent menu. If so, the child menu is removed **174** from the display. If not in the center of the parent menu, a determination **176** is made as to whether the position is over a label menu item. If not, a

determination **178** is made as to whether the cursor is beyond the edge of the radial menu. If it is beyond the edge, a determination **180** is then made as to whether the item has a submenu. If a submenu exists for the item, the submenu is displayed **182** and the submenu is made **184** the current menu.

The pen-up operation as depicted in FIG. **11** performs the item selection determination. The system determines **192** whether the mouse button up event has occurred and if it is up, stops **194** the timer. If the current mode is determined **194** not to be the marking mode, the menus are all erased **196** and the selection is made **198** based on the recorded menu choices. If the mode is the marker mode, the current cursor position and time is recorded **200**. A line is then drawn **202** from the between the previous and current cursor positions. The mark is then erased **204**, the saved positions and times are used to determine **206** the menu choices from the mark and the selection is executed **198**.

It is important that an implementation take certain characteristics of typical window operating systems, with which the invention is typically used, into account as well as accounting for how such operating systems interface with applications. One problem is that many operating systems do not issue an event when a cursor or pointer moved by a pointing device stops moving and the velocity drops to zero. Another problem is that some systems will not relinquish control unless an interrupt is generated. As a result, for such systems the implementation must include a provision for periodically sampling pointing device (cursor) position, such as by including timer interrupts, so that the various steps discussed above will work properly. Some operating systems do not include provisions for saving a portion of a screen into which a menu such as that of the present invention is "popped." In such systems the implementation needs to provide for saving and restoring the portion of the display over written by the pop-up menu when or after the particular selection by the user is made. Because of such problems it is preferred that the invention be implemented in an XWINDOWS system, such as available from Silicon Graphics, Inc.

Additional details concerning the preferred implementation of the process depicted by FIGS. **8** and **11** are set forth in pseudocode included herein as an Appendix. It is preferred that the process as described herein be implemented in a language such as C. However, any language suitable for user interface display generation and processing with a pointing device can be used.

The invention has been described with respect to examples that depict menus with text labels. However, in many cases the preferred mode of display is to use icons. An example of an icon menu that includes radial as well as linear menu items that are arranged in a three-column array is illustrated in FIG. **12**. In this example eight icons **210–224** are shown in the marking menu portion **62** arranged around a center **226** at which a pointer is located. The linear or location selection portion **64** includes six icons **228–238**. This menu is selected in the same way as the text label menu previously described with the user being required to understand the meaning of the icons. The linear menus can be arranged in a two dimensional arrangement as shown in FIG. **12** and the linear menu can be arbitrarily located at any location around the radial menu that is desired depending on where the radial menu actually pops up, something generally decided in the original design of the menu.

FIG. **13** illustrates icon sets **250–264** of a tool box **266** and the expansion of the tool pallet **268** of a selected one of the tool sets **252** with the pointer being positioned at the center of the pallet when expanded allowing continued marking or location dependent selections from the center of the pallet. This illustrates that the pallet can be expanded from a small and generally hard to decipher icon display representation. In addition, the marking selection process can be used to select icons from the small representations of the icons on the left in this figure even when the icons have not been popped-up or expanded to full size, when they might not be "readable" and when the user would have to remember which icon represented which selection.

The menu items shown and discussed herein have included text labels and icons, however, other types of menu items such as moving pictures, symbols, sound, etc. can be substituted. The invention can also be used with any type of menu in a variety of contexts, such as a tool pallet, pulldown menu and object hot spots. The stroke or "link trail" or "rubber band line" also need not be displayed. The number of items in each portion of the menu can vary and the items can be in any desired arrangement.

The many features and advantages of the invention are apparent from the detailed specification and, thus, it is intended by the appended claims to cover all such features and advantages of the invention which fall within the true spirit and scope of the invention. Further, since numerous modifications and changes will readily occur to those skilled in the art, it is not desired to limit the invention to the exact construction and operation illustrated and described, and accordingly all suitable modifications and equivalents may be resorted to, falling within the scope of the invention.

Marking menus with overflow items pseudo-code algorithm.
Assumed system configuration:
Assume we have an event based input system. Three types of events are reported to the application:
1) the mouse button being pressed down
2) the mouse location changing when the mouse button is down
3) the mouse button being released
Assume we have a system capable of scheduling timer interupts with at least a 1/10 of a second frequency.
When events of type 1, 2 and 3 occur the procedures MMpendown, MMpendrag, MMpenup are respectively called. The current mouse x, y location is passed into each of these procedures. When a timer interupt occurs a procedure called MMpenstill is called.
Notes on pseudo-code coventions:
If a subroutine is used more than once, its first occurence in the pseudo code consists of an english description followed by a name for the procedure in brackets. Subsequent uses of the subroutine simply

-continued

use its name.
Some pseudo-code lines are high-level descriptions of more complicated subroutines. These lines may be followed by a name for the routine in brackets. A more detailed pseudo code description for the routine is given later.
Notes of variables. PAUSETIME is a length of time user must hold the mouse still to trigger the display of the menu. In practice PAUSETIME is approximately 1/3 second. JITTEROTERANCE is the amount the mouse can move, in pixels, before our algorthm considers the mouse to no longer be still. In practice JITTERTOLERANCE is approximate 5 pixels.
Comments for explanation are enclosed in by "[ ]"

```
MMpendown(x, y)
        Set the current mode to be MARKING
        Set the current menu to be the root or starting menu
        Schedule a timer interrupt in PAUSETIME (Starttimer)
        Record x, y position as first point in the mark,
                and time this position occured
MMpendrag(x, y)
        if the x,y position has moved a "significant amount" in from
                its last position (CursorMoved?)
            Upschedule the timer interrupt (Stoptimer)
            Starttimer
        if current mode is MARKING
            Save the x, y position, and the time this position occured
            Draw a line segment from previous x,y location to the
                current x,y location
        else
            Hilite the menu item which is associated with the x,y
                location (MenuChoice)
            Record this menu item as being the choice for the current menu.
MMpenstill
        if current mode is MARKING
            Stoptimer
            Set the current mode to MENUING
            replace the mark with menus (ReplaceMark)
            StartTimer
        else
            Stoptimer
            Starttimer
            if current position is in a parent menu center
                Undisplay submenus till you reach the parent menu
                        whose center is being pointed to
            else if the current x,y position is over a menu label or
                        beyond the edge of the current menu radius
                        and the currently selected menu item has a submenu
                Display the submenu of the currently selected item
                Make the submenu the current menu
MMpenup(x,  y)
        Stoptimer
        if current mode is MARKING
            Save the x, y position
            Draw a line segment from previous x,y location
                to the current x,y location
            Remove all drawn line segments from the screen
            Use the saved x,y positions to determine the menu
                choices associated with the mark (SelectionfromMark)
        else
            Remove all the displayed menus from the screen
        Execute the menu choices
MenuChoice
        if the cursor is directly over one of the menu item labels [check both
            the radial portion of the menu and the overflow portion]
            [this step is critical to the algorithm. It makes selection
            when the menu is displayed behave differently from selection
            by drawing a mark. See comment *** below for how selection by
            a mark is determined]
        then the menu choice is that menu item
        else if the cursor is in the center of the radial menu
        then there is no menu choice
        else the choice is the menu item assoicated with the radial
                wedge the cursor is in.
SelectionFromMark
        if the mark is very small (length < size of the center graphic)
        then no selection was made
        if the menu structure is only 1 level deep
        then using the starting point of the mark and the ending point
                determine which radial wedge the endpoint is in. The item
                associated with that wedge is the choosen item
        else
```

-continued

```
                Determine bends and pauses along the marks that
                    correspond to possible transitions from menu selections
                    to submenu selections (GetArticulationPoints)
                From this series of points determine the series of menu item
                    selections made based on the radial wedges only [ *** this
                        is the key to algorithm: marks are analyzed as if
                        no overflow items are present in the menus]
GetArticulationPoints
        Using the list of points and times recorded as the mark was being made,
            find points where the user paused for at least 1/2 second.
        if the number of pause points is greater than the maximum depth the
            menu structure consider the mark a scribble and return no
            articulation points.
        if there are less pause points than the maximum menu depth + 1
            Find the angle changes along the mark that are greater
                than 22.5 degrees [22.5 degrees is 1/2 of 45 degrees which
                        is the angular differences between adjacent wedges.
                        Therefore a change of at least 22.5 degrees in line
                        direction indicates the user meant to select from
                        different wedge in a submenu]
            Use the first maximum menu depth-1 of them as the
                articulation points
ReplaceMark
        GetArticulationPoints
        For each articulation point deterime the series of menu items
                selection based on radial wedge only
        Display those menus each centered at the corresponding
                articulation points
CursorMoved?
        return True if ABS(previous x location - current x location) +
                ABS(previous y location - current y location)
                >JITTERTOLERANCE
```

What is claimed is:

1. A display, comprising:

a radial menu displaying radial menu items from which a selection can be made; and

a linear menu simultaneously displayed with said radial menu having menu items non-hierarchically related to the radial menu items, and said radial and linear menus being simultaneously produced and displayed based on a single input event.

2. A display, comprising:

a radial menu having plural radial menu items from which a selection can be made; and

a linear menu simultaneously displayed with said radial menu, having linear menu items non-hierarchically associated with said radial menu items where a number of linear menu items corresponding to the radial menu items is at least one and less than a total number of the radial menu items, and said radial and linear menus being simultaneously produced and displayed based on a single input event.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO.: 5,926,178
DATED : July 20, 1999
INVENTOR(S): Gordon P. KURTENBACH

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 8, line 37, change "link trail" to --ink trail--.

Signed and Sealed this

Eighth Day of February, 2000

Attest:

Attesting Officer

Q. TODD DICKINSON

Commissioner of Patents and Trademarks

US006433797B1

(12) **United States Patent**
Zellweger

(10) **Patent No.: US 6,433,797 B1**
(45) **Date of Patent: Aug. 13, 2002**

(54) **METHOD AND APPARATUS FOR GENERATING A TAB LIST MENU IN A HIERARCHICAL MENU STRUCTURE**

(76) Inventor: **Paul Zellweger**, 82 Fresh Pond La., Cambridge, MA (US) 02138

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/353,433**

(22) Filed: **Jul. 15, 1999**

Related U.S. Application Data

(60) Provisional application No. 60/095,811, filed on Aug. 4, 1998.

(51) **Int. Cl.**[7] **G06F 3/00**
(52) **U.S. Cl.** **345/762**; 345/825
(58) **Field of Search** ..... 345/700, 744, 345/762, 764, 777, 810, 825, 826, 841, 853, 854, 968; 707/100–102, 103 R, 103 Y, 103, 103 Z, 104.1

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,899,377	A *	2/1990	Bauer et al.	379/354
5,041,967	A *	8/1991	Ephrath et al.	345/825
5,261,042	A *	11/1993	Brandt	345/841
5,483,651	A *	1/1996	Adams et al.	707/1
5,627,980	A *	5/1997	Schilit et al.	345/841
5,630,125	A *	5/1997	Zellweger	707/103
5,745,716	A *	4/1998	Tchao et al.	345/777
5,999,228	A *	12/1999	Matsuura et al.	348/569
6,131,100	A *	10/2000	Zellweger	707/104.1
6,279,005	B1 *	8/2001	Zellweger	707/101

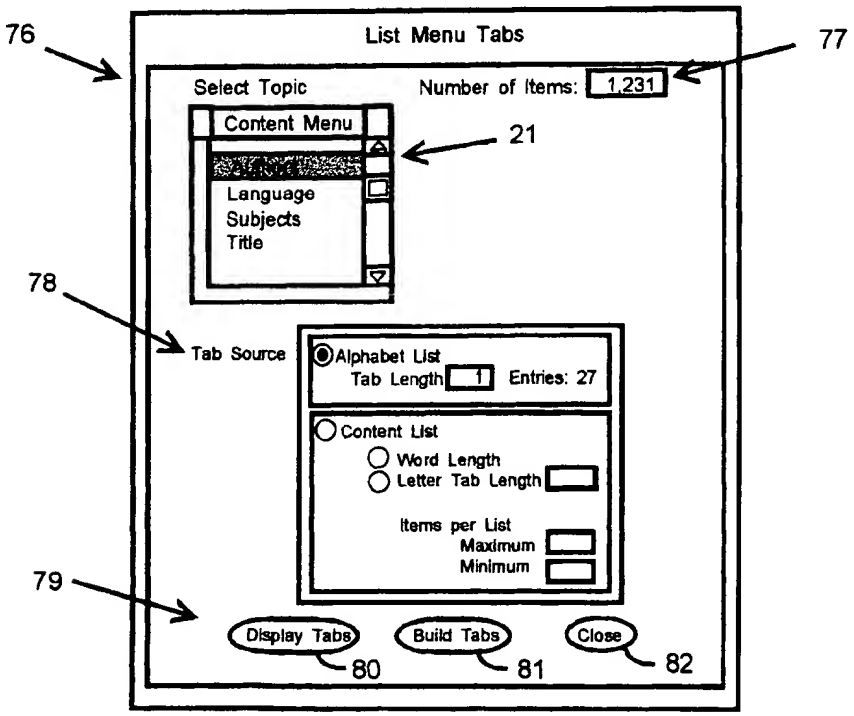
* cited by examiner

Primary Examiner—Crescelle N dela Torre

(57) **ABSTRACT**

Hierarchical menus provide end-users with an easy way to locate information. The underlying hierarchical menu structure is often scalable as there are no limits on the number of entries in a list menu. Yet, this feature is rarely used, if at all, because it can be very impractical for the end-user. As a list menu can contain hundreds, if not thousands of entries, scrolling for a particular entry can be extremely tedious. The present invention discloses a menu authoring system component that enables a developer to generate and build a tab list menu. The tab menu divides a target list menu into tab intervals that can include a specified number of letters or full words. The present invention discloses an interactive configuration window and the program logic that enables a developer to specify settings for these tab menus.

20 Claims, 10 Drawing Sheets



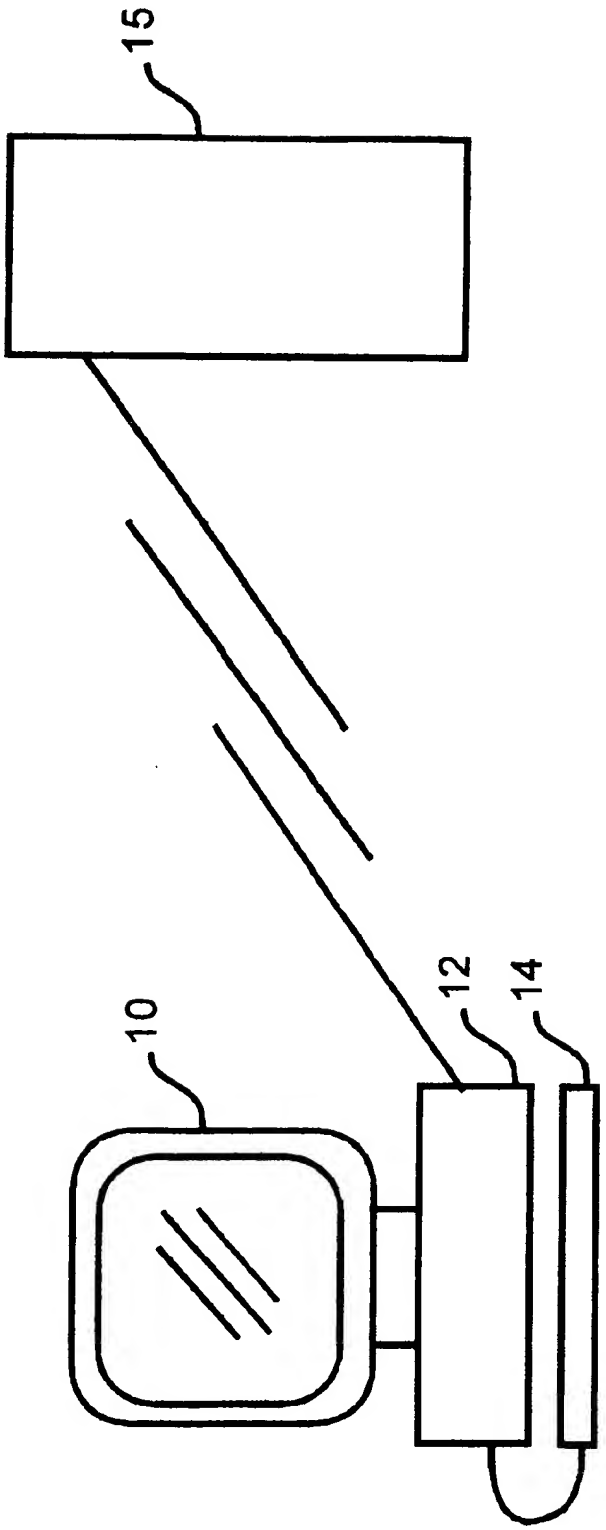
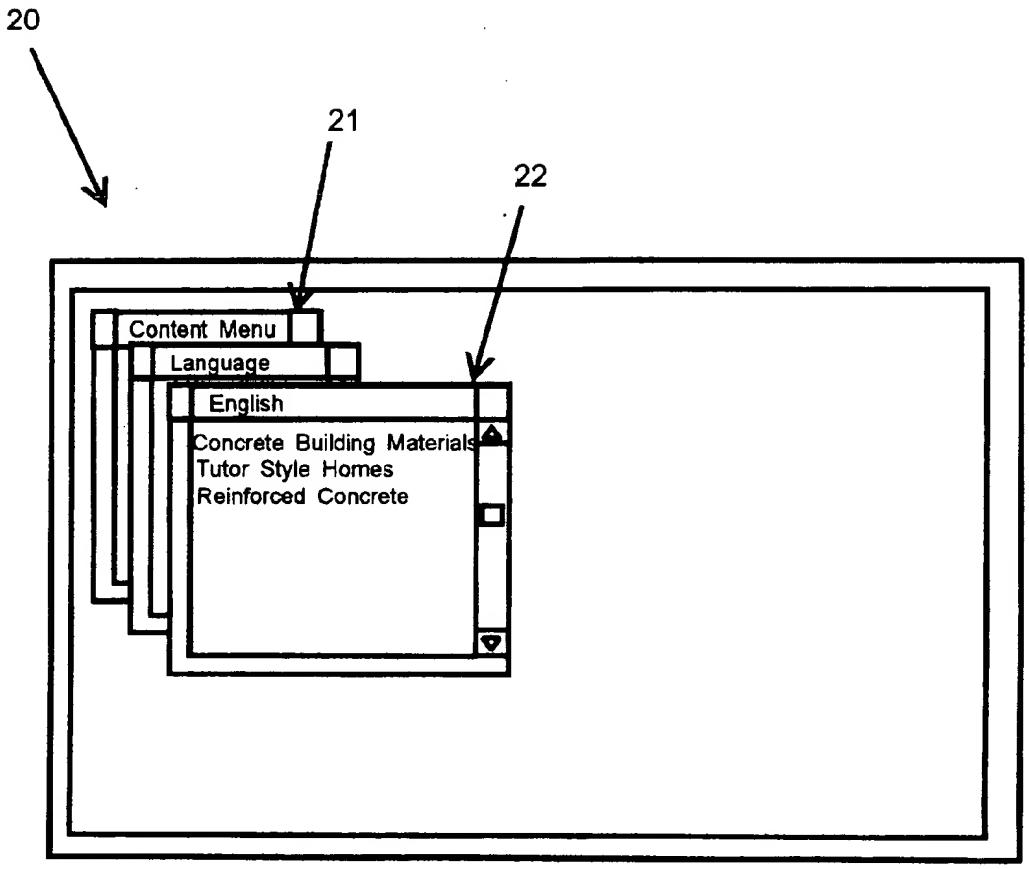
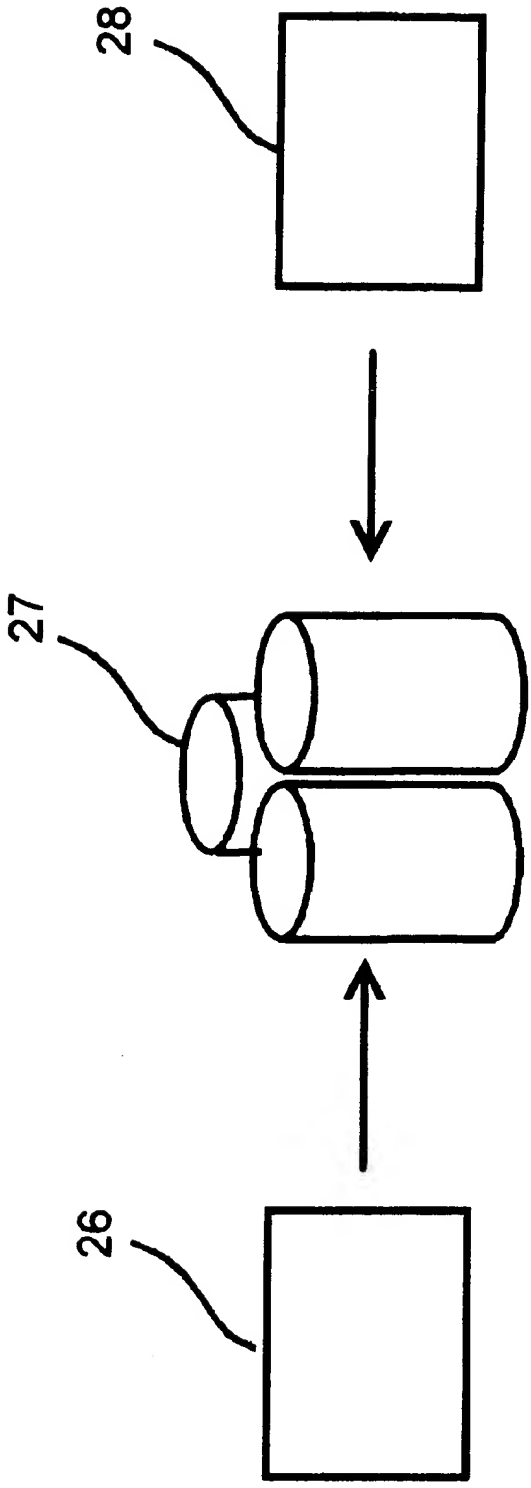


Fig. 1



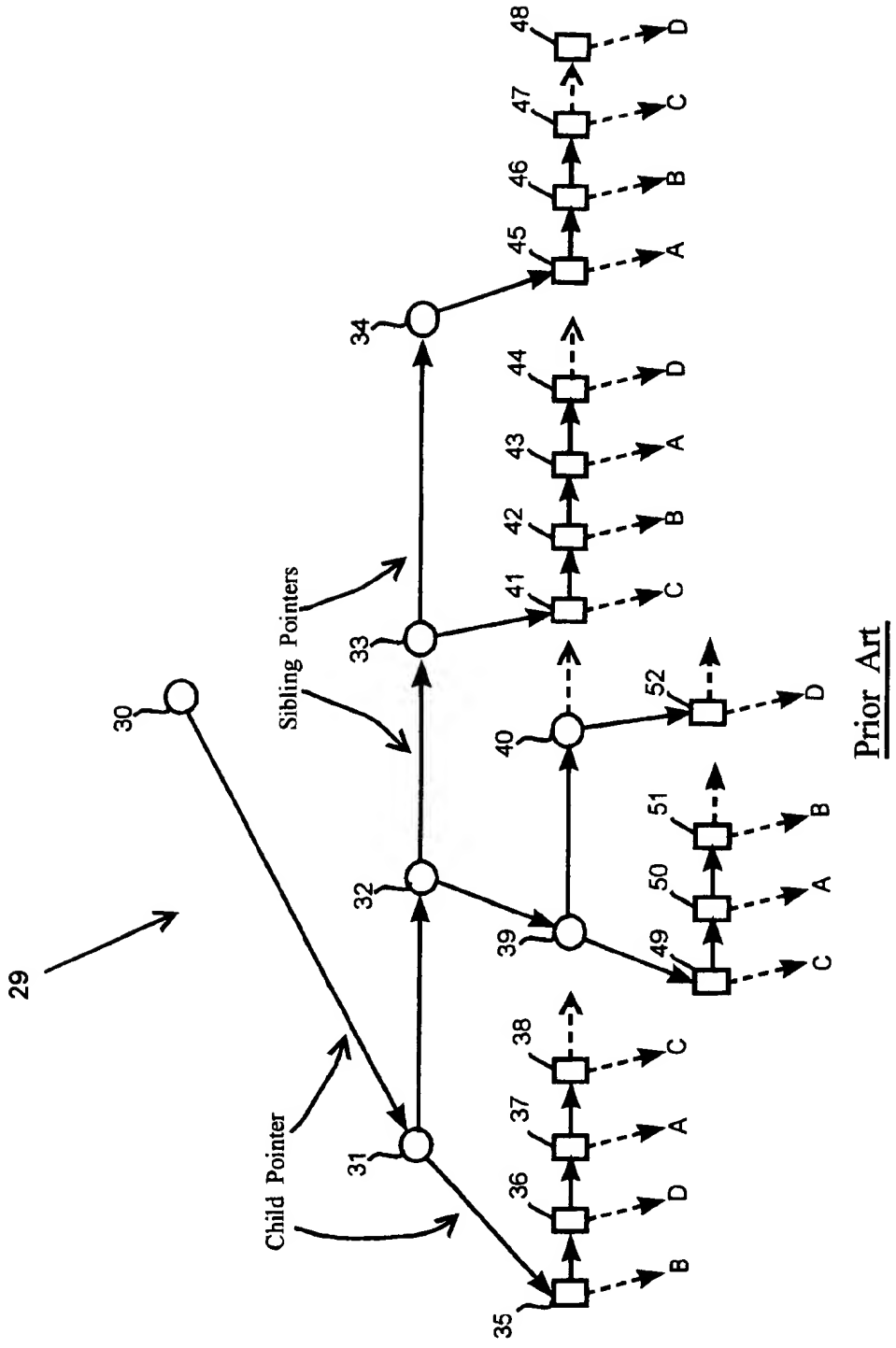
Prior Art

Fig. 2



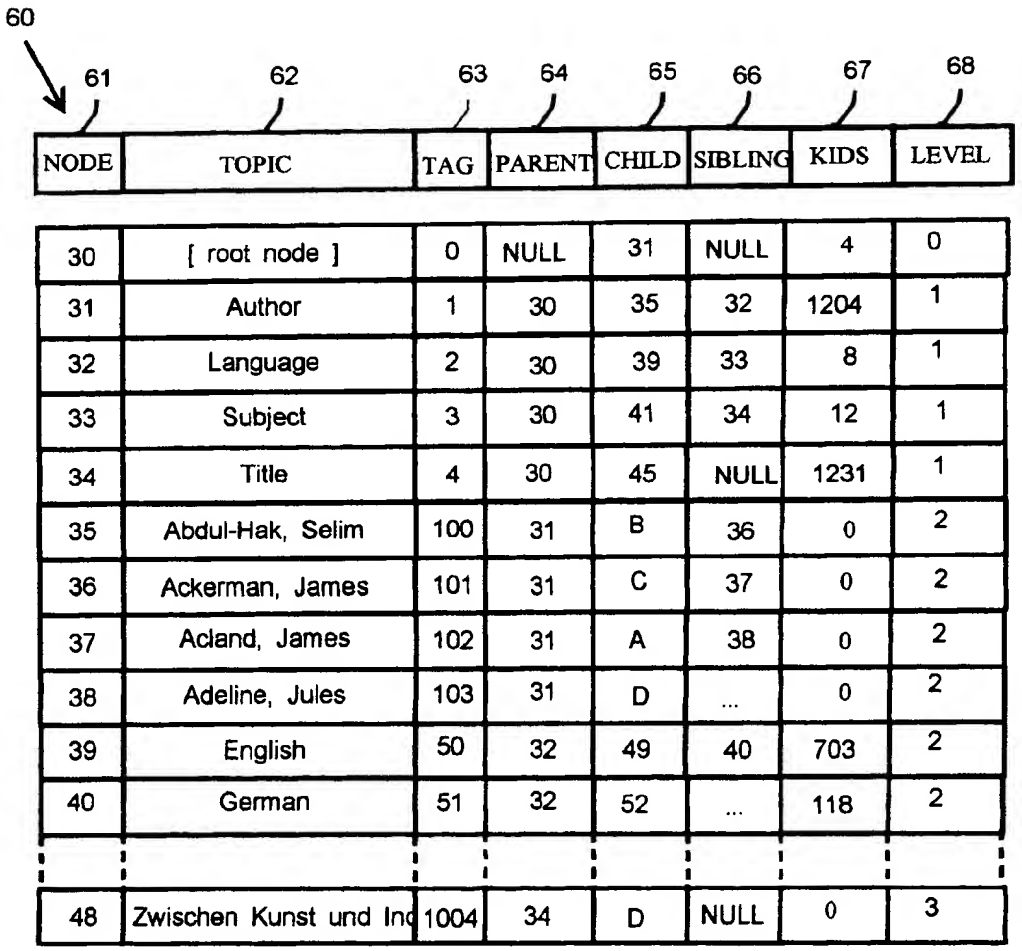
Prior Art

Fig. 3



Prior Art

Fig. 4



NODE	TOPIC	TAG	PARENT	CHILD	SIBLING	KIDS	LEVEL
30	[root node]	0	NULL	31	NULL	4	0
31	Author	1	30	35	32	1204	1
32	Language	2	30	39	33	8	1
33	Subject	3	30	41	34	12	1
34	Title	4	30	45	NULL	1231	1
35	Abdul-Hak, Selim	100	31	B	36	0	2
36	Ackerman, James	101	31	C	37	0	2
37	Acland, James	102	31	A	38	0	2
38	Adeline, Jules	103	31	D	...	0	2
39	English	50	32	49	40	703	2
40	German	51	32	52	...	118	2
48	Zwischen Kunst und Inc	1004	34	D	NULL	0	3

Prior Art

Fig. 5

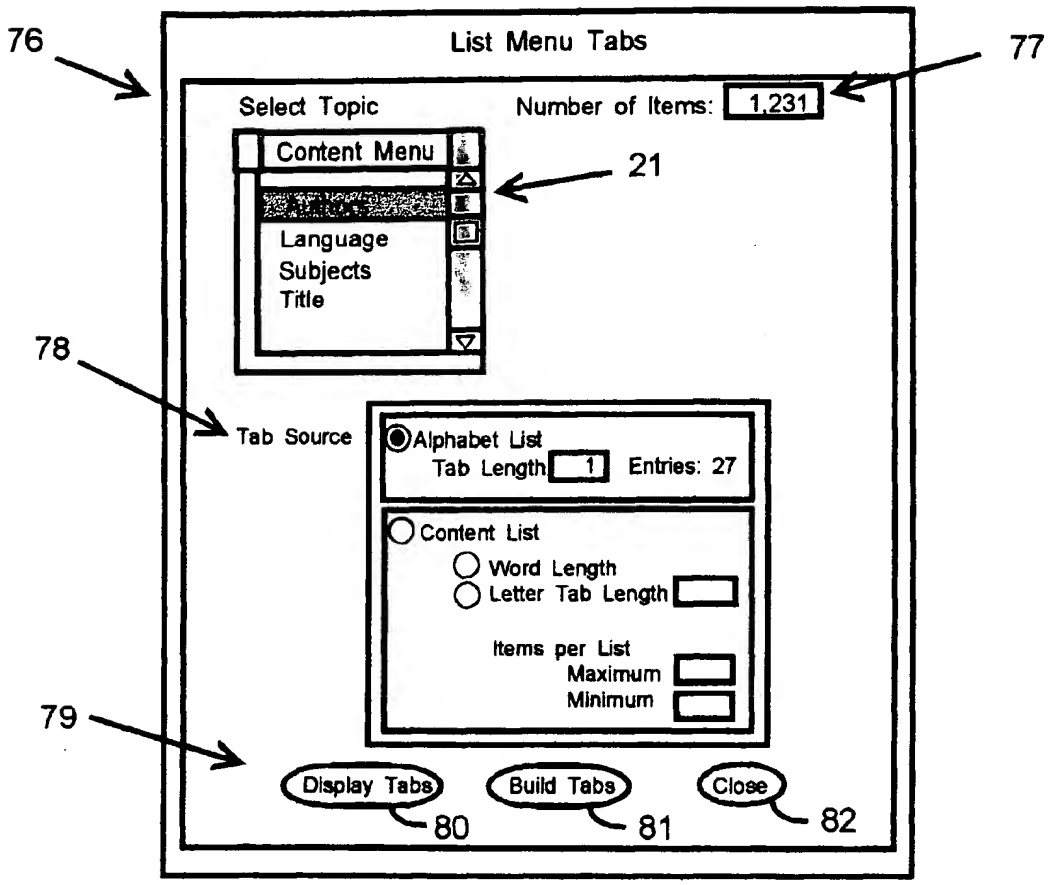

76
List Menu Tabs
77
Select Topic
Number of Items: 1,231
Content Menu
21
Language
Subjects
Title
78
Tab Source
Alphabet List
Tab Length 1 Entries: 27
Content List
Word Length
Letter Tab Length
Items per List
Maximum
Minimum
79
Display Tabs
Build Tabs
Close
80
81
82

Fig. 6a

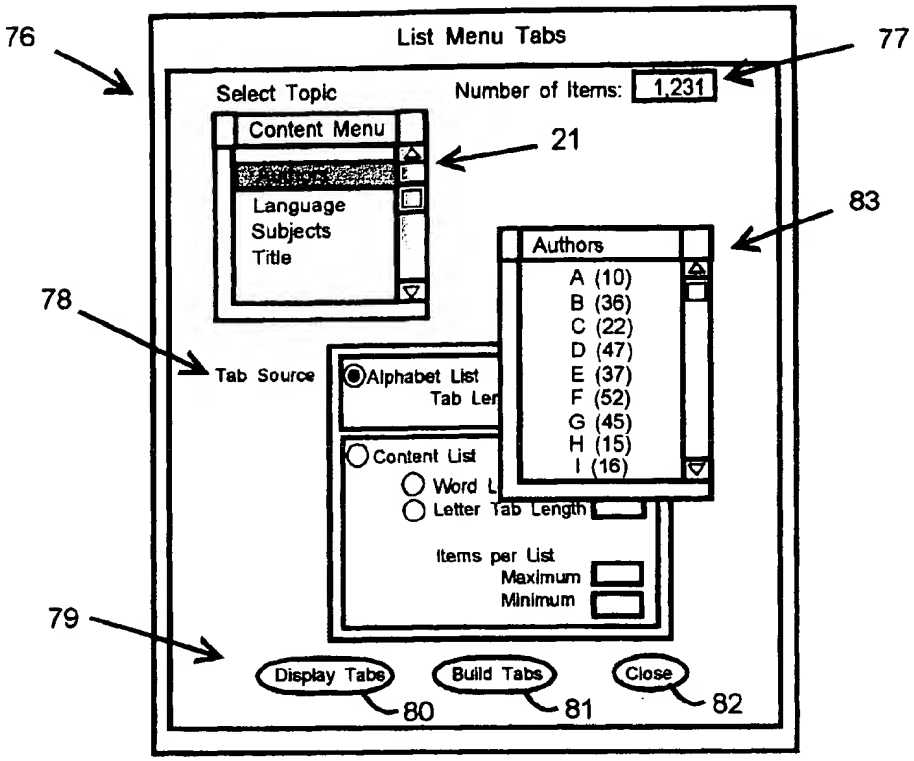

76
List Menu Tabs
77
Select Topic
Number of Items: 1,231
Content Menu
21
Language
Subjects
Title
83
Authors
A (10)
B (36)
C (22)
D (47)
E (37)
F (52)
G (45)
H (15)
I (16)
78
Tab Source
Alphabet List
Tab Len
Content List
Word L
Letter Tab Length
Items per List
Maximum
Minimum
79
Display Tabs
Build Tabs
Close
80
81
82

Fig. 6b

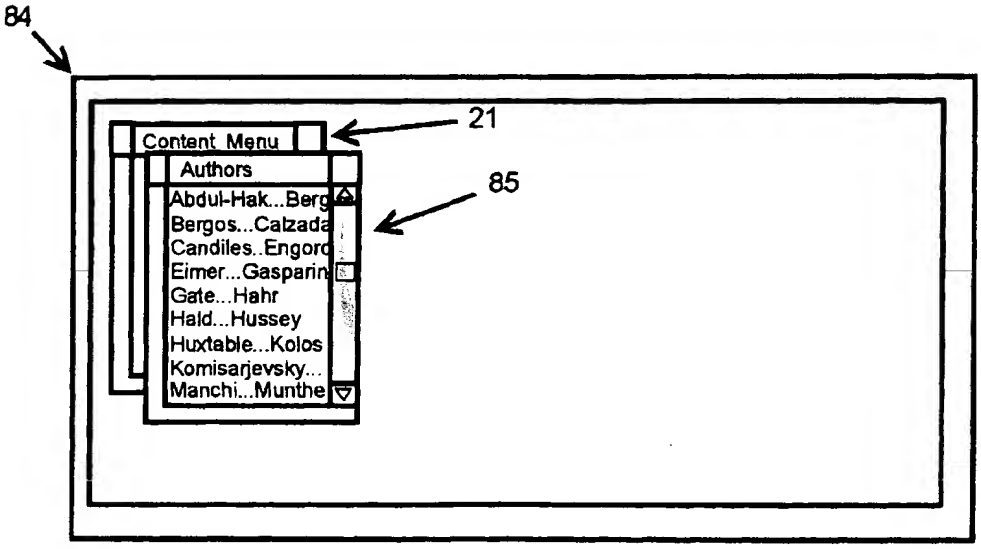

84
Content Menu
21
Authors
85
Abdul-Hak...Berg
Bergos...Calzada
Candiles..Engord
Eimer...Gasparin
Gate...Hahr
Hald...Hussey
Huxtable...Kolos
Komisarjevsky...
Manchi...Munthe

Fig. 7

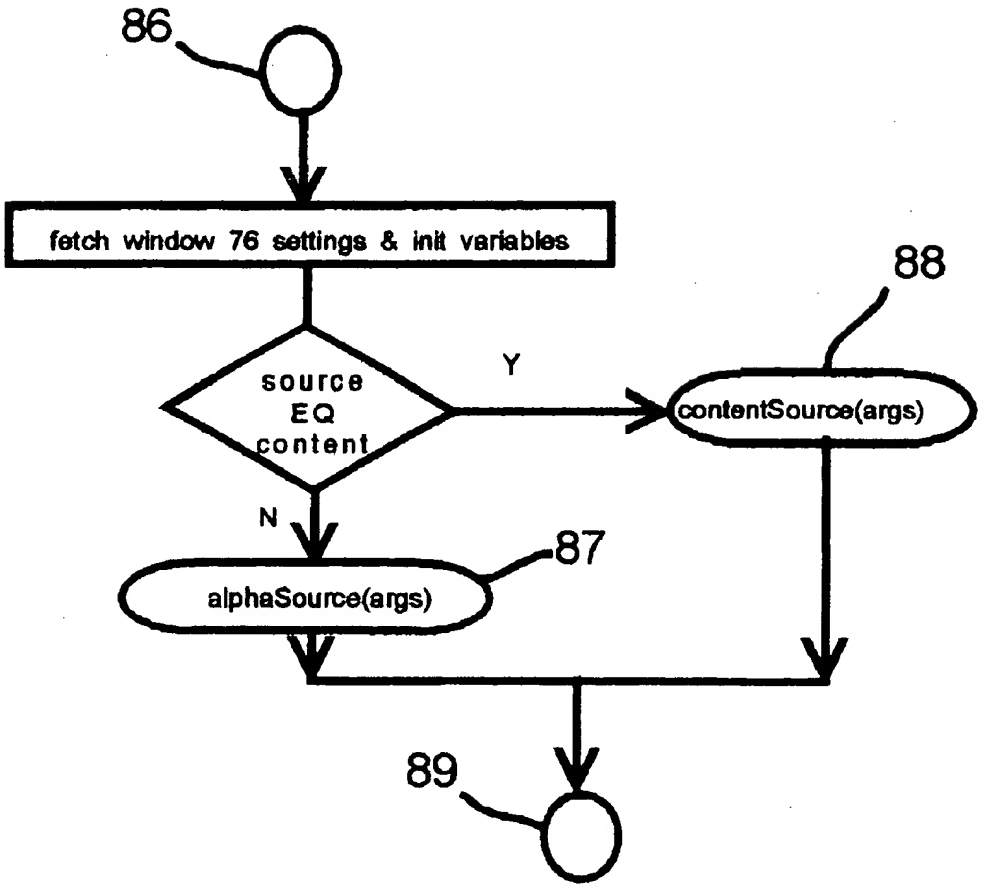

86
fetch window 76 settings & init variables
source EQ content
Y
88
contentSource(args)
N
87
alphaSource(args)
89

Fig. 8a

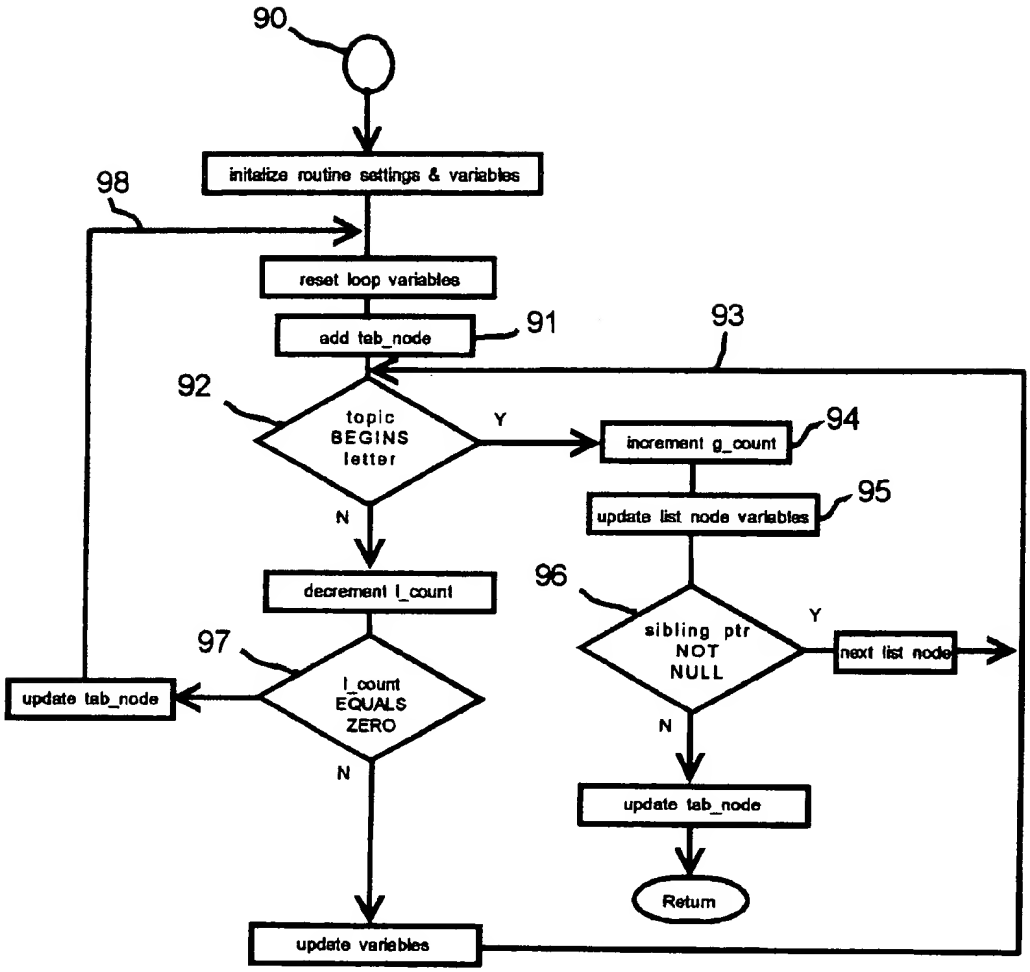

90
initalize routine settings & variables
98
reset loop variables
add tab_node
91
93
92
topic BEGINS letter
Y
increment g_count
94
update list node variables
95
N
decrement l_count
96
sibling ptr NOT NULL
Y
next list node
97
l_count EQUALS ZERO
update tab_node
N
update tab_node
Return
update variables

Fig. 8b

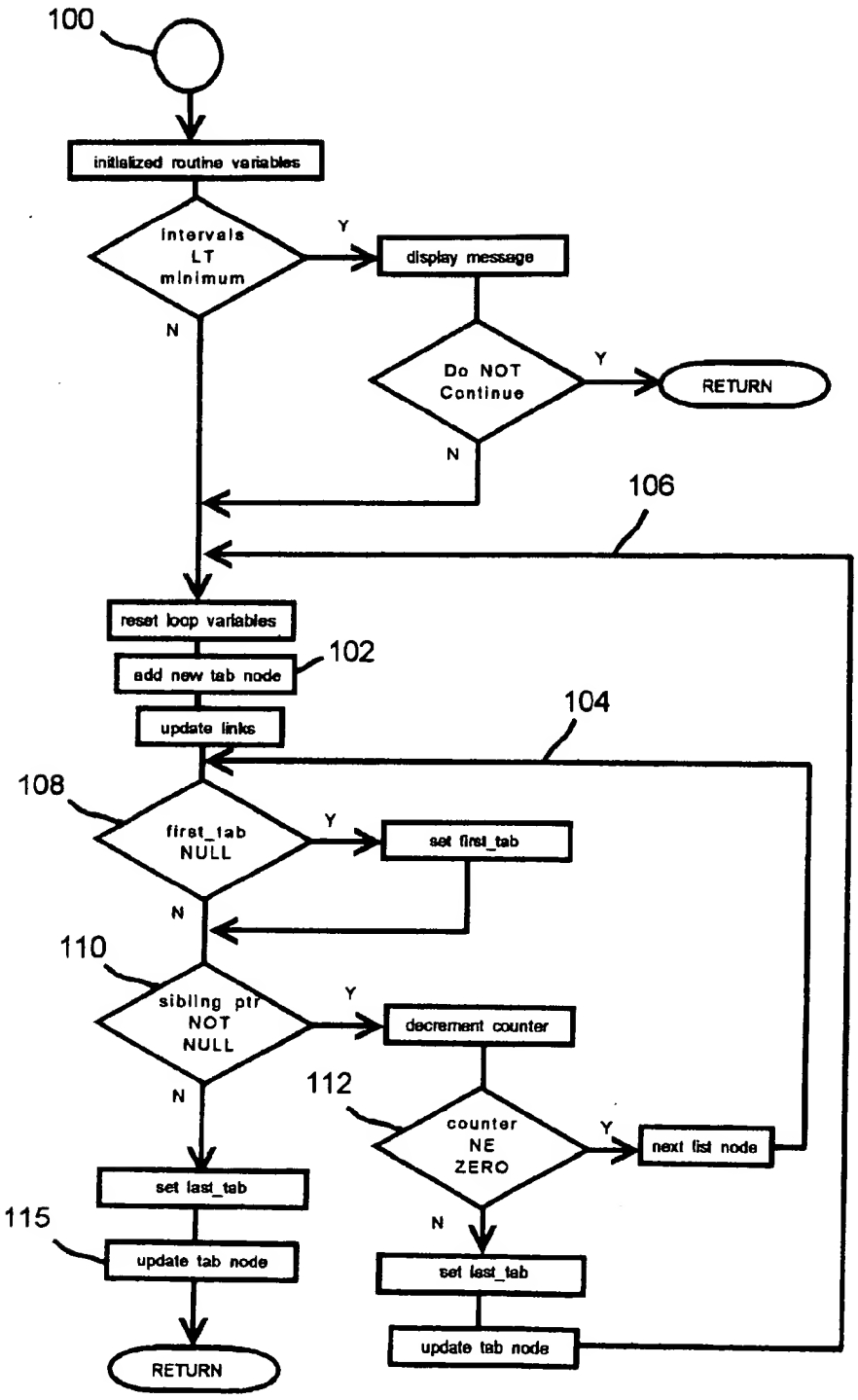

100
initialized routine variables
intervals LT minimum
Y
display message
N
Do NOT Continue
Y
RETURN
N
106
reset loop variables
add new tab node
102
update links
104
108
first_tab NULL
Y
set first_tab
N
110
sibling ptr NOT NULL
Y
decrement counter
N
112
counter NE ZERO
Y
next list node
N
set last_tab
115
update tab node
RETURN
set last_tab
update tab node

Fig. 8c

METHOD AND APPARATUS FOR GENERATING A TAB LIST MENU IN A HIERARCHICAL MENU STRUCTURE

CROSS REFERENCE TO RELATED APPLICATION

This application is based on Provisional Application No. 60/095,811 entitled Method and Apparatus for Generating Tabs in a Content Menu Using an Open Hierarchical Data Structure, filed Aug. 4, 1998 abandoned.

REFERENCES

U.S. Patent DOCUMENTS		
6,279,005	8//01	Zellweger
6,131,100	10/00	Zellweger
5,999,228	12/99	Matsuura et al
5,745,716	4/28/98	Tchao et. al.
5,630,125	5/13/97	Zellweger
5,627,980	5/97	Schilit et al
5,483,651	1/96	Adama et al
5,261,042	11/93	Brandt
5,041,967	8/91	Ephrath et al
4,899,377	2/90	Bauer et al

Other References

Zellweger, Paul. "Web-based Sales: Defining the Cognitive Buyer." *International Journal of Electronic Markets. Vol. 7 No. 3, 1997. pp. 16–22.*

Zellweger, Paul. The Knowledge Tree. unpublished paper, copyright ArborWay, Inc. Cambridge, Mass., Jan. 3, 1998.

FIELD OF INVENTION

The present invention relates to computer software used to build and maintain a menu system and, more specifically, it relates to a menu management system that enables a developer to generate a tab list menu and link it to an existing menu system.

BACKGROUND

A hierarchical menu system organizes information under a series of nested topics, so end-users can navigate down its lists to the reach information they need. Help systems represent, perhaps, the most prevalent and easily recognized application of this type of menu. Yet, prior to Zellweger (U.S. Pat. No. 5,630,125), menu paths in these systems have always been mutually exclusive because all prior menu authoring system art was based on the binary search tree. Zellweger's authoring system and its underlying data structure, the open hierarchical data structure art, on the other hand, enables multiple paths to reach the same information.

In a menu authoring system the underlying data structure organizes information and arranges its descriptive topics into a series of nested lists and paths. This includes data structures like the binary search tree, as well as Zellweger's more flexible structure. When the menu developer is ready to build a hierarchical menu system, the data structure, again either a search tree or Zellweger's structure, transforms these lists of nested topic lists into menu data used by the display system.

As these data structures are often scalable, there is no limit on the number of items in a topic list. Yet, these lists never grew too large, until U.S. Pat. Nos. 6,279,005 & 6,131,100 by Zellweger, because all data entry was made by hand. Zellweger's disclosures show how to automatically generate networks of lists and paths from external sources of data such as database structures and specially prepared files. These methods automatically propagate Zellweger's structure and thereby overcome a prior obstacle, expensive and labor intensive data entry.

Mapping database values into a menu data structure in a automatic fashion creates new challenges for its menu end-users, especially when these list menus can get exceptionally large. Previously, the number of entries in each list menu was under a degree of control because it was always built by hand. Automation removes this level of control, and produces list menus that can contain hundreds, if not thousands of entries that make scrolling for a particular entry extremely tedious. To combat this problem, the present invention discloses ways to automate the generation of a tab list menu that in many respects, corresponds to a thumbnail tab in a book, its paper counterpart.

Tchao et. al (Apr. 28, 1998 U.S. Pat. No. 5,745,716) discloses a type of electronic tab displayed on a monitor of a hand-held computer device. Each tab represents a subset of items in a larger list. The user interface includes a display of the complete set of tabs and the capability of selecting a tab and cycling through items corresponding to the selected sublist. In contrast, the present invention is dedicated to a totally different type of user interface, nested list menus in a content menu, and a different retrieval objective, organizing information to link directly to each object.

SUMMARY OF THE INVENTION

The present invention discloses a menu authoring system that generates a tab list menu that links to a set of sublists which, previously, represented one long continuous list of menu items. With the present invention the menu developer has full control over how the original list is divided into intervals and how the tab menu list portray these divisions. Using an interactive configuration window the developer selects the source for these divisions, an alphabet list or the content itself, and can display the tab list menu prior to making permanent changes to the menu structure. The menu developer is free to select any list in a menu structure as a candidate for a tab list menu. When the new tab list menu is generated the present invention links this menu to the prior incoming and outgoing list objects.

BRIEF DESCRIPTION OF THE FIGURES

FIG. **1** depicts the network apparatus of the present invention.

FIG. **2** depicts a graphical representation of the prior art of the content menu user interface.

FIG. **3** depicts the three major software components of the prior art of the content menu.

FIG. **4** depicts a graphical representation of the prior art of the open hierarchical data structure.

FIG. **5** depicts the prior art of the database structure used to represent node information in the open hierarchical data structure.

FIGS. **6***a* and **6***b* depict the interactive user interface of the configuration window in the present invention employed by a developer to generate a tab list menu.

FIG. **7** depicts a graphical representation of a tab list menu generated by the present invention.

FIGS. **8***a* through **8***c* depict the flow chart of the program logic used to generate a tab list menu.

DETAILED DESCRIPTION OF THE FIGURES

To illustrate the present invention a menu system representing a collection of books will be used. At the upper-most

level of the menu system four major topics are presented: authors, languages, subject, and titles. Under languages and subjects, topics links to a one or more sublists that narrow the subject material. This network of topics represents the primary distinctive feature of the content menu, allowing multiple ways of locating the same thing.

FIG. **1** depicts the desktop computer apparatus **12** of the present invention that includes a monitor device **10** and an input keyboard device **14**. In one embodiment of the present invention a menu developer builds content menu tabs of the present invention on computer **12** in a stand-alone setting. In another embodiment of the present invention the developer to builds a tab list menu on a server computer **15** from client computer **12**.

A graphic representation of the prior art of the user interface known as a content menu **21** is depicted in FIG. **2**. Monitor **10** displays the content window **20**. The content menu **21** includes one or more nested list menus. Each list menu **22** in **21** includes at least one menu entry, a menu title, and a scrolling device if needed. Each time the end-user selects an entry in **22** that content menu **21** either displays the next list menu **22** or an information object at the end of a menu path. Note, that from the end-user's perspective, content menu **21** works just like any other hierarchical menu system.

The three primary software components of the prior menu art are depicted in FIG. **3**. This includes an authoring system **26**, menu files **27**, and software components **28** that access menu files **27** and display content menu **21**.

The authoring system **26** represents the software used to build and maintain a hierarchical menu structure and to generate menu data files **27**. Advances brought about by the present invention represent major improvements to the prior art of authoring system **26**.

The open hierarchical data structure **29**, depicted in FIG. **4**, represents a specific type of hierarchical menu structure art. It includes a single root node **30**, one or more internal nodes known as branching nodes such as node **31**, and one or more data nodes that link to an information object such as node **48**. Generational terms, such as parent child, and sibling, are used to describe elements in structure **29** and how they relate to each other. Once again, the distinctive feature of **29** is that it allows more than one paths to reach an information object at the bottom of the structure.

Each node in structure **29** has a outgoing arc or child pointer that flows either to a child or an information object. As a child node can be located anywhere below root **30** in structure **29**, each nonroot node can have more than one parent node that includes a primary and one or more step-parents.

Each node in structure has a topic or an information key that corresponds to a entry in a list menu. The sibling pointer produces a sibling list that corresponds to a list menu. A child pointer links a list menu entry to an offspring list menu or to an information object.

Structure **29** can represent complex hierarchical networks that enable multiple paths to reach the same information object, as well as mutually exclusive pathways found in conventional menu structures. Therefore, alternative embodiments of the present invention include applications of program logic that can be applied to any structure that can be depicted by **29**, including a binary search tree, the hierarchical structure typically associated with nested list menus.

The preferred teachings of the prior art use elements in database structure **60**, depicted in FIG. **5**, to represent information about each node in structure **29**. This includes node id **61**, topic in **62**, child pointer **65**, and sibling pointer **66**. The database structure **60** also includes an end-user tag in **63**, a pointer to a primary parent in **64**, the number of children nodes in kids **67**, and the node's level in the structure in **68**.

In the present invention, authoring system **26** also includes configuration window **76**, depicted in FIG. **6**, that enables a menu developer to specify tab list menu settings and generate the tab list menu for content menu **21**. First, he or she navigates content menu **21** in window **76** to select a menu entry that points to a target list menu. Note that any list menu in **29** can be a target list menu including a tab list menu. Each time an entry in **21** is highlighted, the number of items in its successor or offspring list is displayed in field **77**.

After the developer selects a target list, he or she selects a radio button in **78** to designate the source for the tab list menu. This can be either an Alphabetical List or the target list menu, identified as a Content List.

Under the Alphabetical List option in **78** the developer can select the length of each tab. In the example given in FIG. **6***a* the length is set to one. This means there would be a single letter in each tab and there would be 27 entries in the list.

Under the Content List option in **78** the developer can specify a full word or length of letters for the first and last elements in a tab interval. In addition, the developer can specify the maximum and minimum number of entries in each tab list menu.

In region **79** of configuration window **76** the developer can display the tab list menu with button **80**, make permanent changes to the menu structure with **81**, and close window **76** with **82**.

FIG. **6***b* depicts configuration window **76** after the Display Tabs button **80** was selected by the developer. In this example of the interactive capabilities of the present invention **76** generates and displays a sample tab list menu **83**, from the settings in FIG. **6***a*, an alphabet list source having a single letter. Next to each entry in **83** is the number of elements associated with the alphabet letter interval. Note that the sample tab list menu does not alter an existing menu structure.

Window **84**, in FIG. **7**, depicts content menu **21** after a developer selected Build Tabs button **81** in window **76**. Tab list menu **85** in **84** was built using the authors content list with a word length. Each tab entry in **85** depicts an interval using the author's last name as the first and last elements. So when an end-user selects the "Abdul-Hak . . . Berg" entry, the content menu generates a new list menu that starts with "Abdul-Hak" and ends with "Berg".

Finally, FIGS. **8***a* through **8***c* depict an embodiment of the program logic in the present invention used to generate a tab list menu in a menu structure like an open hierarchical data structure.

Routine **86**, depicted in FIG. **8***a*, is an event-driven routine that is executed when Build Tabs button **81** is selected in configuration window **76**. It depicts the main program driver that generates tab list menus for content menu **21**. After fetching selected options from window **76** and initializing routine variables routine **86** calls either contentSource at **88** or alphaSource at **89**. Upon return, control flows back to window **76** at **89**.

The program logic of alphaSource routine **90** is depicted by FIG. **8***b*. Program control flows from routine **86** at **89** to

90. Routine 90 generates a tab list menu based on the alphabet. The number of letters in each tab is determined by the developer in window 76 in the Alphabet List source box in region 78.

In loop 93, routine 90 iterates through nodes in the target list to identify elements associated with the current alphabet tab. At 91 routine 90 adds a new tab entry in the tab list menu. Each time a member in the target list is encountered g_count is incremented at 94; this counter is used to display the number of elements in the offspring list. At 95, links in the target list are updated and the next sibling node is made current. When the sibling pointer is identified as null at 96, indicating the end of a target list, routine 90 updates the current tab node and returns to the calling routine.

Loop 98, controls the number of letters associated with each tab interval. Each time the first letter in a target node's topic is new, l_count is decremented. At 97, routine 90 checks l_count, the alphabet letter sequence length designated by the developer, to divert flow to loop 93 to continue with the same tab or 98 to start a new one, So a single letter setting would generate a new tab_node for each letter in the alphabet, and a four letter tab would generate ABCD, EFGH, and so on.

And lastly, contentSource routine 100, depicted in FIG. 8c, generates a tab list menu based on the target list menu and the interval size set by the developer. Upon entry, 100 initializes routine variables that calculate the interval size, handle topic sting construction, and manage links to update the target node list and the new tab list menu. Next, routine 100 checks the interval size and prompts the developer to continue if it is less than the designated minimum.

At the top of loop 106 routine 100 adds a new tab entry to the tab list in the menu structure. Loop 106 uses counter, a loop variable set to the calculated interval size, to iterate through the correct number of nodes in the target list. On the first pass it sets first_tab using the topic of the first node in the interval and the letter sequence size specified in window 76. If the current node has a sibling pointer the loop decrements counter. Each time the counter is greater than zero loop 104 uses the sibling pointer to fetch the next list node.

When the counter reaches zero program control flows to loop 106 and last_tab is set to the topic in the current node. Next loop 106 uses first_tab and last_tab to construct a tab topic and add it to the current tab node. Flow now s tarts back at the top of loop 106 where loop variables get reset and a new tab node is added to the menu structure.

At 110, routine 100 checks the current node in the target list for a null sibling pointer, signaling the last node. A null pointer causes program control to construct and add the last tab topic and return to the calling routine.

Conclusion

This concludes the description of an embodiment of the invention. The foregoing description of the embodiment of the invention has been presented for the purpose of illustration and description. It is not intended to be exhaustive or limit the invention to the precise form disclosed. Many modifications and variations are possible in light of the above teaching. The scope of the present invention is not intended to be limited by this detailed description, but rather by the claims appended hereto.

This concludes the description of an embodiment of the invention. The foregoing description of the embodiment of the invention has been presented for the purpose of illustration and description. It is not intended to be exhaustive or limit the invention to the precise form disclosed. Many modifications and variations are possible in light of the above teaching. The scope of the present invention is not intended to be limited by this detailed description, but rather by the claims appended hereto.

I claim:

1. A menu authoring system that enables a developer to generate a content-based tab list menu consisting of,

displaying a tab list configuration interface,

fetching a selection made by said developer from said tab list configuration interface that designates a target list in a hierarchical menu structure as a source of tab labels for said content-based tab list menu,

fetching a setting made by said developer from said tab list configuration interface that represents a size factor in determining a sublist in said target list,

creating said sublist from said target list in said hierarchical menu structure,

creating a tab label from content in said sublist,

creating said content-based tab list menu based on said tab label,

linking said tab label in said content-based tab list menu to said sublist in said hierarchical menu structure.

2. The menu authoring system of claim 1 wherein is implemented in a computer program language that is compatible with at least one computer operating system.

3. The menu authoring system of claim 1 further includes displaying in said tab list configuration interface a navigation interface that enables said developer to navigate said hierarchical menu structure to select said target list.

4. The menu authoring system of claim 1 further includes displaying in said tab list configuration interface the number of entries in said target list menu in said hierarchical menu structure.

5. The menu authoring system of claim 1 further includes displaying in said tab list configuration interface an interface that enables said developer to specify different display options when creating said tab label.

6. The menu authoring system of claim 1 further includes displaying in said tab list configuration interface an interface that enables said developer to specify said factor size as a maximum number of items in said sublist.

7. The menu authoring system of claim 1 further includes displaying in said tab list configuration interface an interface that enables said developer to specify said size factor as a minimum number of items in said sublist.

8. The menu authoring system of claim 1 further includes the option of generating a tab list menu based on the alphabet sequence.

9. The menu authoring system of claim 1 further includes displaying the number of items associated with said tab label in said content-based tab list menu.

10. The menu authoring system of claim 1 further includes displaying in said tab list configuration interface said content-based tab list menu.

11. A menu authoring system that enables a developer to generate a tab list menu consisting of,

the means for displaying a tab list configuration interface,

the means for fetching a selection made by said developer from said tab list configuration interface that designates a target list in a hierarchical menu structure as a source of tab labels for said content-based tab list menu,

the means for fetching a setting made by said developer from said tab list configuration interface that represents a size factor in determining a sublist in said target list,

the means for creating said sublist from said target list in said hierarchical menu structure,

the means for creating a tab label from content in said sublist,

the means for creating said content-based tab list menu based on said tab label,

the means for linking said tab label in said content-based tab list menu to said sublist in said hierarchical menu structure.

12. The menu authoring system of claim **11** wherein is implemented in a computer program language that is compatible with at least one computer operating system.

13. The menu authoring system of claim **11** further includes the means for displaying in said tab list configuration interface a navigation interface that enables said developer to navigate said hierarchical menu structure to select said target list.

14. The menu authoring system of claim **11** further includes the means for displaying in said tab list configuration interface the number of entries in said target list menu in said hierarchical menu structure.

15. The menu authoring system of claim **11** further includes the means for displaying in said tab list configuration interface an interface that enables said developer to specify different display options when creating said tab label.

16. The menu authoring system of claim **11** further includes the means for displaying in said tab list configuration interface an interface that enables said developer to specify said factor size as a maximum number of items in said sublist.

17. The menu authoring system of claim **11** further includes the means for displaying in said tab list configuration interface an interface that enables said developer to specify said size factor as a minimum number of items in said sublist.

18. The menu authoring system of claim **11** further includes the means for generating a tab list menu based on the alphabet sequence.

19. The menu authoring system of claim **11** further includes the means for displaying the number of items associated with said tab label in said content-based tab list menu.

20. The menu authoring system of claim **11** further includes the means for displaying in said tab list configuration interface said content-based tab list menu.

* * * * *

US005704051A

United States Patent [19]

Lane et al.

[11] Patent Number: 5,704,051

[45] Date of Patent: Dec. 30, 1997

[54] **HIERARCHICAL MENU BAR SYSTEM WITH DYNAMIC GRAPHICS AND TEXT WINDOWS**

[76] Inventors: **Ronald S. Lane; Miriam Weiss Lane**, both of 3 Eagle View Ct., Monsey, N.Y. 10952

[21] Appl. No.: **613,527**

[22] Filed: **Mar. 11, 1996**

Related U.S. Application Data

[63] Continuation of Ser. No. 155,464, Nov. 19, 1993, abandoned.

[51] **Int. Cl.**[6] **G06F 3/00**

[52] **U.S. Cl.** **395/357**; 395/353

[58] **Field of Search** 395/326-358, 395/806-807; 345/146, 123-125, 902, 117-120

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,648,062	3/1987	Johnson et al.	395/334
4,710,763	1/1987	Franke et al.	395/353
4,712,191	12/1987	Penna	395/353
5,122,972	6/1992	Richards et al.	395/353 X
5,295,242	3/1994	Mashruwala et al.	395/353

OTHER PUBLICATIONS

WordPerfect for Windows, WordPerfect Corp., May 1993, screen pp. 1-2.

Mossberg, "Parental Guilt Sells Encyclopedias on CD-ROM, Too", Wall Street Journal, Apr. 29, 1993, p. B1 (W).

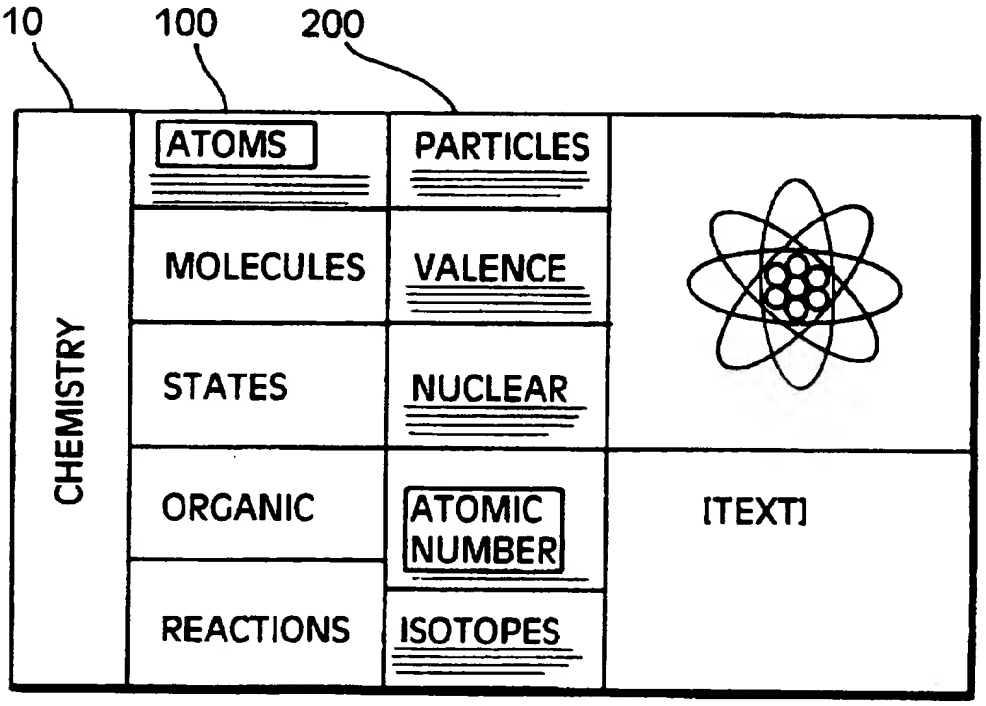
Primary Examiner—John E. Breene

[57] **ABSTRACT**

A data processor for managing a multilevel application wherein the data processor includes a display controller that creates a three-level menu window and a data window. The menu window includes at least two levels that each incorporate select commands associated with discrete subjects wherein menu commands are concurrently displayed on screen to provide historical access information. Menu commands are converted into display presentations where each level defines a greater degree of information detail on a given subject. The data display window for these presentations is further divided into windows for text and graphics.

The foregoing arrangement has been found to be exceptionally effective at providing educational or tutorial information access in a efficient manner.

2 Claims, 3 Drawing Sheets



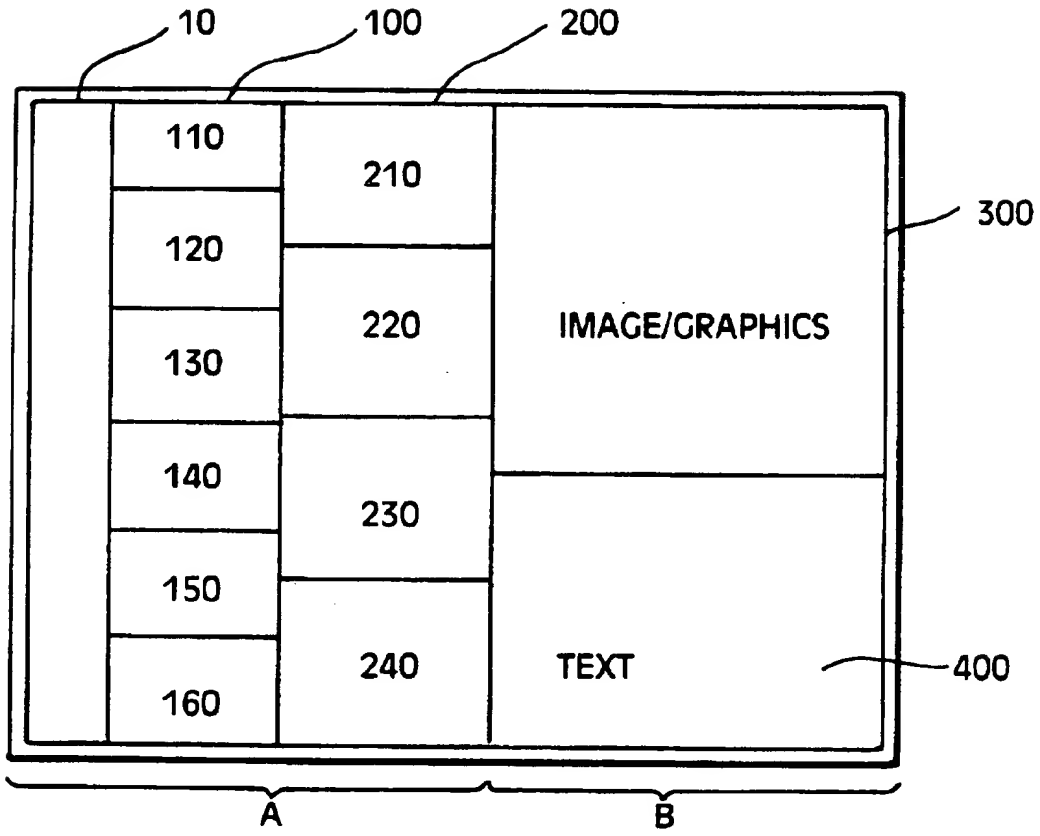

10
100
200
110
120
130
140
150
160
210
220
230
240
IMAGE/GRAPHICS
TEXT
300
400
A
B

FIG. 1

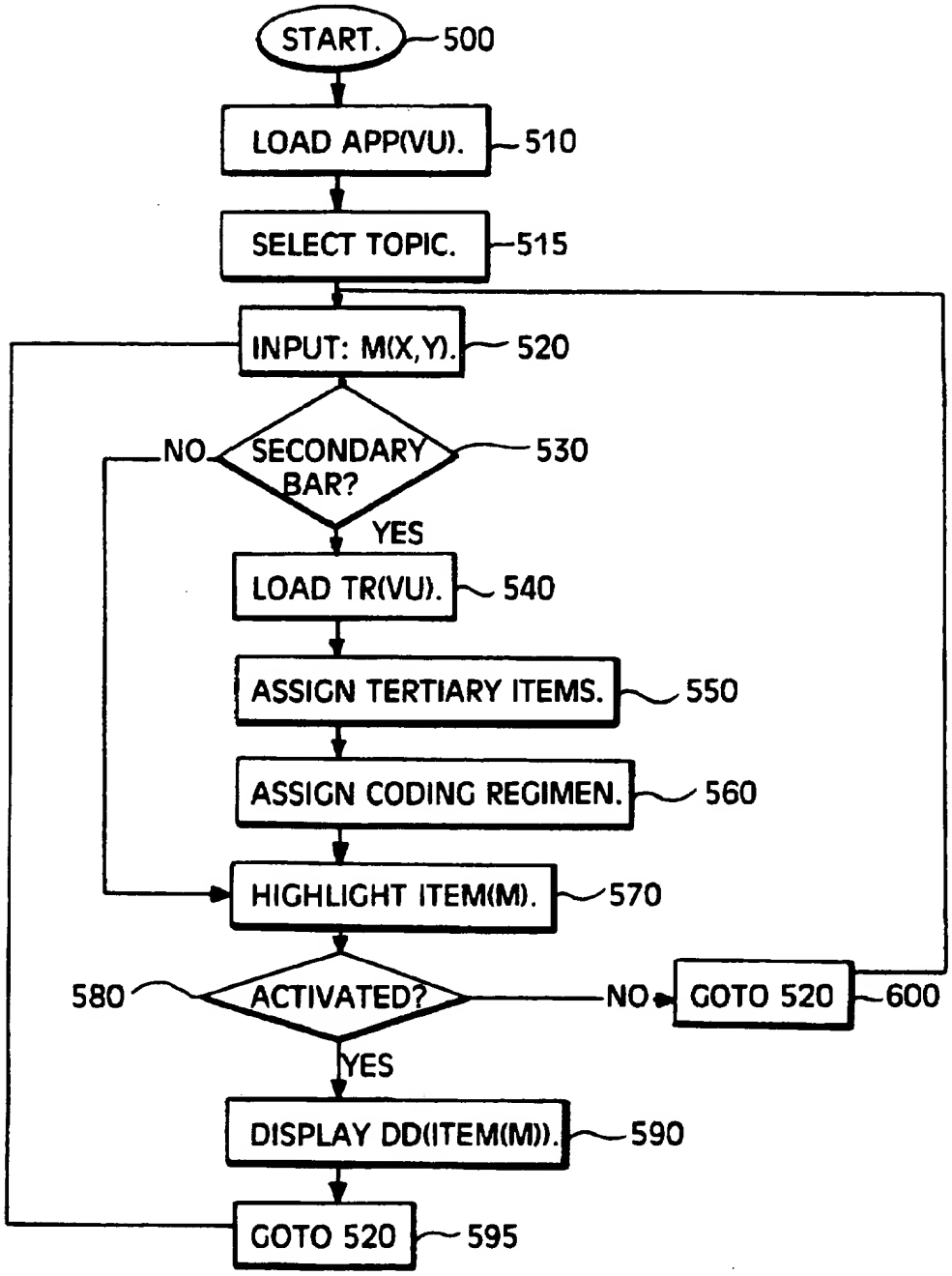

START.
500
LOAD APP(VU).
510
SELECT TOPIC.
515
INPUT: M(X,Y).
520
NO
SECONDARY BAR?
530
YES
LOAD TR(VU).
540
ASSIGN TERTIARY ITEMS.
550
ASSIGN CODING REGIMEN.
560
HIGHLIGHT ITEM(M).
570
580
ACTIVATED?
NO
GOTO 520
600
YES
DISPLAY DD(ITEM(M)).
590
GOTO 520
595

FIG. 2

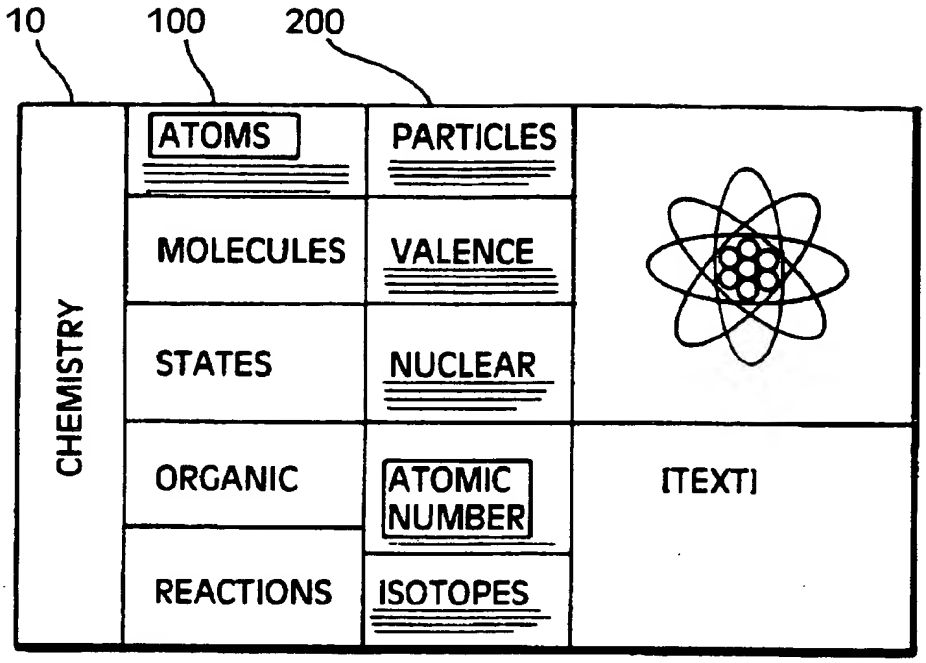

10
100
200
CHEMISTRY
ATOMS
PARTICLES
MOLECULES
VALENCE
STATES
NUCLEAR
ORGANIC
ATOMIC NUMBER
REACTIONS
ISOTOPES
[TEXT]

FIG. 3

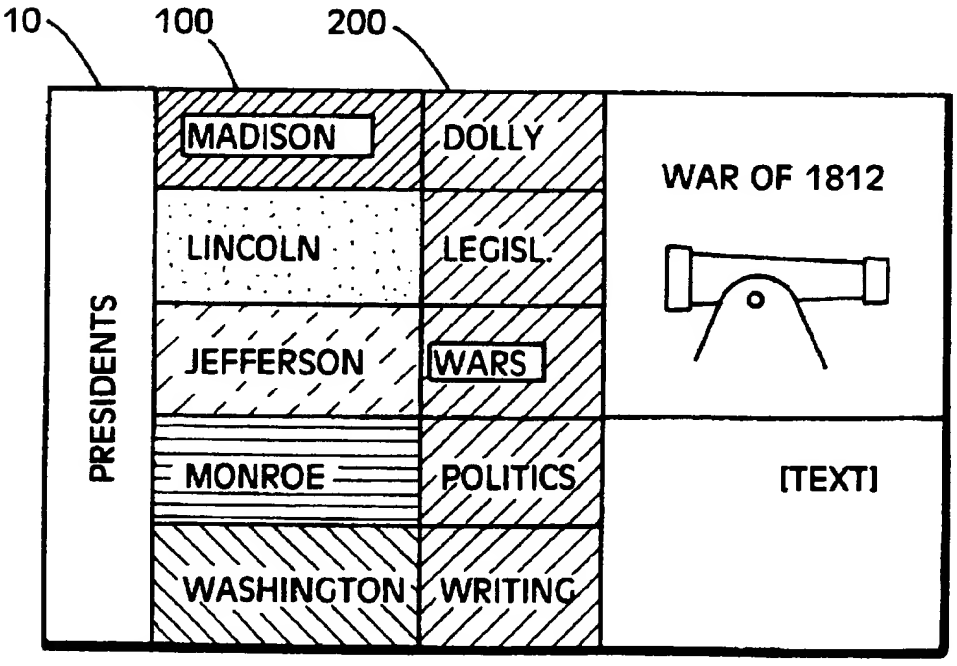

10
100
200
PRESIDENTS
MADISON
DOLLY
WAR OF 1812
LINCOLN
LEGISL.
JEFFERSON
WARS
MONROE
POLITICS
[TEXT]
WASHINGTON
WRITING

FIG. 4

HIERARCHICAL MENU BAR SYSTEM WITH DYNAMIC GRAPHICS AND TEXT WINDOWS

This application is a continuation of Ser. No. 08/155,464, filed Nov. 19, 1993, now abandoned.

The present invention relates generally to data processing systems enhanced by dynamic information access and, more particularly, a data process methodology that provides a dynamic multilayer menuing system for layered access to complex information in a user friendly manner.

BACKGROUND OF THE INVENTION

Computer systems may differ in many ways, but most have a few select components that are common among the systems. For example, most computers have data storage, via magnetic memory, data input through a keyboard and/or position transducer (mouse), and finally data output via screen display (CRT, LCD or other). The present invention is primarily concerned with the display system, and, in particular, the manner in which the two-dimensional display provides information about seeking data in systems that have multiple levels of detail.

Information that is visually displayed by a computer is formatted into a matrix, wherein the matrix defines the two-dimensional mapping of color pixels on a display screen. The arrangement and orientation of these individually activated pixels provides larger visual structures representative of the underlying data of interest. The visual structures can be based on graphics and/or text. In either case, the system includes program controlled subroutines that properly organizes the visual structures into the graphics and/or text that convey information to the user in a meaningful way.

Of course, no computer system can provide all the data about multiple levels of information to the user in one screen display. Therefore, there has developed within the software community a variety of distinct menuing systems that gives the user a plethora of paths to specific information. The differing systems range in complexity, with many employing sequential screens where different choices are displayed contingent on selections made from previous screen menus.

For example, in text-based systems, menus are presented on the display giving the user a list of selections, e.g., "file"; "edit"; etc. The user selects one, wherein the original menu is removed and replaced with a second menu giving a subset of new options to the user, e.g., "save" or "delete". Usually, the new menu replaces the old one, thereby removing the lineage of the menu selection. In a graphical interface, such as a windows environment, many menus are presented together; but when the operation is selected, each menu disappears—removing menu path lineage information.

For simple applications involving few choices, the lack of lineage information is mostly unimportant. But for many sophisticated applications, there is a need for a menuing system that provides a lineage of the path taken to access the presently displayed data. This is particularly important for applications designed for operation by the less computer literate.

Moreover, the initial menus in a multi-tier menu system are provided as merely a portal for deeper probing into the application choices. Conventionally, there was no link between available data and the initial menu entry. Only when the most detailed and lowest echelon of the menu selections is reached will the system provide information on the subjects of interest to the user, specifically tied to the final level of the menu. This, of course, denies the user information related, not only to the overall structure of the search, but also that global, overview and higher level information often of critical interest in seeing the "bigger picture" that puts the finer detail information within a context.

This is particularly true when dealing with educational software systems. It is now well accepted that computers offer a powerful tool for use in education of adolescents and adults. One of the major criticisms of the use of computers in formal education stems from their use as purely an "answer" machine—answers given without the attendant methods of more sophisticated exploration normally associated with information acquisition. For example, computer access to information on whales would involve simply entering the key word "whale" and then displaying a segment from memory on this subject. This segment may provide information on the whale, its habitats, and its relationship with other water borne mammals. However, the system's key word format fails to provide information to the user about where whales fit with the related data available in the information base. Generally, this format denies the user the details on the structure of information associated with a given topic, and thus fails to provide context and perspective to the chosen topics. This is a significant limitation as the understanding of information structure derived from global and/or more detailed views, and the knowledge of interrelationships between subjects is often more critical in education than the brute obtainment of facts.

Similar issues arise outside the field of computer assisted education and can be seen in other applications relating to business, entertainment, utilities and shopping. In each, existing menuing systems are, for the most part, designed for users who know where they want to go, and reaching this destination is the primary if not sole function of the menu. The use of the menu per se to give access to the underlying information at any and all levels of detail is simply ignored.

It is, therefore, recognized that, especially in the field of education, but also in many other fields, current menuing techniques starve the user regarding important information about broader views of the subject in the interest of reaching the final, most detailed display information. The present invention is directed to a solution to this problem.

OBJECTS AND SUMMARY OF THE PRESENT INVENTION

It is an object of the present invention to provide a data processing system that enhances the menuing interface for accessing information in the separate levels of the application being implemented.

It is yet another object of the present invention to provide a user menu system that is linked to a data display screen so that data associated with a given menu choice is currently displayed to the user at whatever level in the menu structure is selected by the user.

It is a further object of the present invention to provide an educational data processor having a multilayer access system where succeeding menus are retained on the display, so that the overall structure of the application remains visible to the user.

It is yet another object of the present invention to provide a data processor that iteratively accesses increasing levels of detail regarding a selected topic pursuant to user selected menu entries wherein the data displayed is based on user menu selection and corresponds to information having a degree of detail commensurate with the chosen level within the menu hierarchy.

It is another object of the present invention to provide an enhanced user interface for a data processing system wherein the hierarchial path to the final level in a multilayered application remains on the display providing historical data regarding the path to the current level of the application.

The above and other objects of the present invention are realized in a data processing interface integrated with a hierarchial menuing system that provides multilevel menu structure as a continuous aspect of the screen display. The screen display is subdivided into a first data display segment window for the presentation of content information relating to a selected menu category item, and a second menu display segment for the presentation of menu item and path information, tracing the path to the current display of content information.

An exemplary arrangement of the menu display segment involves a primary categorical display bar on a single subject, contiguous with a secondary display bar having a plurality of secondary items representing selections corresponding to the primary categorical display bar, but providing a greater level of detail within that subject. This secondary display bar is contiguous with a tertiary display bar, which is composed of tertiary items. These tertiary items relate to an item from the secondary bar that is selected by the user and provide even greater detail regarding the subject matter of the secondary item selected.

The system accesses presentation data in accordance with the selection of items within the menu structure. This information is then presented in the display segment of the screen. Importantly, the information accessed and presented reflects the level of detail of the item selected from the menu. For example, selecting an item from the primary bar will produce a broad or global presentation on that subject matter in the data display segment. As the user penetrates to lower menu bars, selected items will produce display information on that same subject area, but with increasing detail.

Navigation through the menu bars and their associated levels is further enhanced by an item coding regimen wherein secondary items are displayed with differing and distinct colors and/or patterns. Selection of a secondary item controls the content and appearance of the tertiary bar—all tertiary bar items will be related to the selected secondary item and will be presented in the tertiary bar having the same coding, e.g., the same color or pattern of the selected secondary item. If additional bars extend beyond the tertiary bar, these will follow this regimen using pattern or color coding to relate lower level items to the selected tertiary item. Patterns will include hash marks, dots, etc.

In accordance with the varying aspects of the present invention, the data display segment of the screen is further subdivided into two separate data windows. The first window is for graphics and image display and the second window is for display of text associated with the graphics display.

The foregoing features of the present invention may be more fully appreciated from the following detailed description of a specific illustrative example thereof, presented hereinbelow in conjunction with the accompanying drawing of which:

DESCRIPTION OF THE FIGURES

FIG. 1 is a diagram of a representative screen display for use in conjunction with the present invention;

FIG. 2 is a logic flow chart for controlling system logic that defines menu attributes;

FIG. 3 is a representative screen display corresponding to the system of the present invention; and

FIG. 4 is a representative screen display corresponding to the system of the present invention.

DETAILED DESCRIPTION OF THE PRESENT INVENTION

First briefly in overview, the present invention concerns the user interface in implementing a specific data processing application. The inventive interface is characterized by a unique menuing system that permits layered access to information via selection of a hierarchy of items described and represented by contiguous menu bars. The menuing system concurrently displays at least three levels of data access with each level of the menu supplying information to the user tied to a specific degree of detail contingent on the item and menu bar selected. The levels are dynamic and may be addressed at any time by the user to shift level and current data focus. Separately, the resulting data is shown on a split screen combining text and images on the subject matter pertaining to the user's selections.

In metaphorical terms, the inventive memory system operates like a microscope with three different magnification levels. The first magnification operates to provide a macro view of the subject. For example, if examining a bacteria cell, the first magnification would show the entire cell and its position in relation to other cells. Shifting to the second (higher) magnification brings into view the interior of the cell, depicting cell compounds such as the nucleus and mitochondria. And, finally, the third (highest) magnification provides even more detail on the interior of the nucleus, such as chromosomes. However, isolation to this final view denies the user of the microscope information about the cell in broader terms, e.g., cell wall, nucleus position, etc. Thus, the microscope is a more powerful tool when capable of viewing the subject at multiple levels that vary in scope. Similarly, information acquisition is of greater benefit when accessed using a menuing system providing multiple levels of data access varying in degrees of detail.

With the above brief overview in mind, attention is now directed to FIG. 1 which presents a diagram of a typical display using the inventive system. More particularly, FIG. 1 depicts a two-dimensional representation of a computer output screen, conventional in a hardware sense and corresponding to any number of display systems on the market that vary in such parameters as dot pitch size, resolution, scanning rate, refresh rate, etc.

The screen of FIG. 1 is selectively divided into two major segments, indicated as segment A, the menu profile including menu bars **10**, **100** and **200**, and segment B, the data content display window. Dimensionally, the menu segment A will consume approximately 30% of the total screen area with the remaining 70% used for the data segment B.

Continuing with FIG. 1, the menu bars of the menu segment A represent succeeding levels of access and detail through the menuing system. More particularly, primary bar **10** is the highest current level and provides a global categorization of the data. For example, in an educational system on the presidents of the United States, primary bar **10** would be labeled "Presidents"—thus indicating throughout system use the general subject matter for this segment of the application.

In this context, user selection of the primary bar would trigger memory access of the data display file corresponding to the general subject matter of the primary bar **10**. If this subject matter is titled "Presidents", the data display window would then provide a general presentation on this subject, e.g., an image of the White House connected to images of

Congress and the Supreme Court and possibly some text on the origins of Presidential power from the theories of Locke and Hobbes.

Secondary Bar **100** is logically linked to Primary bar **10** as a sub-category thereof, providing the next level of detail about the primary bar **10** subject. Secondary Bar **100** is further divided into secondary items **110–160**. In this fashion, each secondary item represents a logical information sub-category to the subject matter defined by primary bar **10** and provides a more detailed examination of this topic.

For example, if primary bar **10** is directed to Presidents, secondary bar **100** would be directed to specific Presidents and the secondary items **110–160** would be e.g., "Washington", "Jefferson", "Madison", "Monroe", etc., wherein each secondary item represents a sub-category within the broad subject defined by primary bar **10**. Selection of any of these secondary items will access a data file containing graphical and textual information on that item's subject for display, and further define the menu items in the tertiary bar.

Following this pattern, the tertiary (TR) bar **200** represents an even more detailed level of information which is contingent on the selection of one of the items from the secondary bar. Expressed somewhat differently, the selection of an item in the secondary bar will define the subject categories available for the items of the next level, which are then placed in the tertiary bar **200**—as each item in the secondary bar is logically concatenated to a series of additional items representing deeper, more probing presentations available in the tertiary level. The selected secondary bar item's color or pattern will be carried forward to correspond to the concatenated tertiary items, thereby indicating to the user the connection between the subject matter of the two levels.

For example, if "Jefferson" is selected from the items in the secondary bar, the system processor will access the general file from memory on Jefferson, which is then conveyed to the data display segment B. This general presentation on Jefferson might include biographical text with pictures which may be leisurely reviewed by the user. In addition, system processor pulls from memory selected items associated with Jefferson and displays these as tertiary items in tertiary bar **200** as e.g., "Monticello", "Supreme Court Selections", "Louisiana Purchase", etc. These tertiary items are color coded to correspond to the same color as the Jefferson secondary item and differentiated from the other secondary items such as Washington and Monroe which have different colors.

Continuing with this process, if the user selects an item from the tertiary bar, e.g., "Monticello", the system uses this selection to access a data file on the subject Monticello. The general data on Jefferson is then removed from the data display by the processor and replaced with the more detailed information on Monticello in the data display segment B.

The menu items displayed in tertiary bar **200** contain only items related to one of the items in the secondary bar **100**, in this case Jefferson. In addition, the items in tertiary bar **200** contain the same color or pattern, and match the color or pattern of the item selected in secondary bar **100**.

Importantly, when the final data file is accessed for display in the data segment B, all of the menus used to access this data are shown in full in the menu segment A, i.e., Bars **10**, **100** and **200** remain on the screen giving the user a detailed presentation not only of the final data, but on the path and method required to access this data. The user may retreat to higher levels of information by merely selecting items in the higher level menu bars. This gives the user a much fuller appreciation of select information, as this information is set within a framework of the general subject area. It also enhances data accessing skills as the user reasons through the matrix of levels to obtain the desired result.

Referring to the data segment B of FIG. **1**, this window is further divided into text and image/graphic/video portions, wherein the data accessed for display includes both images and graphics combined with text for a more complete and thorough presentation. Continuing with the example discussed above, the "Monticello" group of the third level accesses from system memory data on Monticello—which may include text data—with multiple pages accessed individually or through per se page scrolling techniques available in the text partition of the data display segment. The Monticello file may also include pictures of the famous estate, a short video (with accompanying audio) charts and graphs, etc.—each displayable in the image/graphics partition of the data segment B. The menu, occupying approximately 30% of the screen, may be "minimalized" at any time allowing the data portion of the screen to expand and fill the entire screen. The continued existence of the menu portion may be represented by a small pictorial "icon" in the expanded data display. Selecting this icon, via a mouse click, will regenerate the menu portion of the display to its full and original form.

Turning now to FIG. **2**, a logic flow chart is presented delineating system operation. Logic conceptually begins at Start block **500**, and continues to block **510** wherein the display controlling application APP(VU) specific to a selected topic of interest is loaded into system memory. At block **515**, the user enters a general subject area which defines the data base accessed by the processor via menu commands as translated by the user. Pursuant to this entry, the system generates the primary and secondary bar structures for display and provides the plurality of secondary items linked to the selected primary bar subject. The tertiary bar and the data display windows default initially to blank screens and remain blank until the activation of an item in the primary and/or secondary bar. This completes the initial screen display.

Once activated, the system orchestrates the menu bar displays in accordance with the position of the mouse pointing device. This is initiated at block **520**, wherein the position of the mouse on the screen is ascertained in matrix terms, M(X,Y).

Once the mouse position is entered, test **530** thereafter determines whether the mouse is located in the secondary bar; if so, logic proceeds to block **540** where the corresponding tertiary items are accessed pursuant to the above mouse position X,Y coordinates. The system then selectively divides the tertiary bar into the individual tertiary items—each with a title relating to its more detailed subject area of interest, block **550**. Logic then proceeds to block **560** wherein the system assigns visual enhancements, e.g., color or pattern attributes, to the tertiary items culled from memory to correspond to the selected secondary item color or pattern.

At block **570**, the mouse position is identified within the border of an item matrix in a bar and the system then tests whether a command has been separately entered by the user at that position. In this context, a command refers to a user request to access a data file associated with the item at the activated mouse position for display in the data window. This selection is made by activating the mouse button, test

580. If activated, logic proceeds to block 590 and the visual and text data linked to the selected item is sent to the display. Logic loops back to block 520 to ascertain changes in mouse location in (X, Y) and new item actuation. Operation continues through selective iterations as presented by the user.

EXAMPLE

The foregoing features of the invention are presented in a system directed to an educational program (application). In this example, the user is given an initial screen menu representing a variety of topics. Exemplary topics, for purposes of this example, include "U.S. Presidents" and "Chemistry Principles".

User selection of Presidents drives system logic to display the initial screen in accordance with the present invention (FIG. 4). In particular, three bar menus are displayed on the right third of the screen as follows: The primary bar will be assigned the general topic "Presidents", bar 10 in FIG. 1. The system accesses the secondary items associated with "Presidents" e.g., the five items on Washington, Jefferson, Madison, Monroe, Lincoln, and assigns these items to the secondary bar 100. At this time, the user may access data for display on either the general category of Presidents, or any of the secondary items by selection, via mouse.

Although the items of the tertiary bar are originally blank, user selection of Madison in the secondary bar will trigger memory access of the tertiary items associated with Madison for display. The system subdivides the tertiary bar into rectangles equalling the number of tertiary items associated with Madison and then assigns these items to the tertiary bar 200, i.e., the five tertiary items pertaining to Madison—Dolly, Legislation, Wars, Politics and Writings.

The data display takes up the remaining two-thirds of the screen. At this point, the user may optionally select entries relating to Madison from the tertiary bar 200. If "Wars" is chosen, data is presented on the data screen relating to the significant conflicts of Madison's administration, including pictures, maps, etc., with connecting text, video and voice/music audio output (if on a multi-media system).

If the user moves the cursor to the second level item "Jefferson", the five groups relating to Madison are removed from the third level and replaced with topics on Jefferson, such as "Monticello", etc.

The user may exit from Presidents and choose "Chemistry Principles", wherein a separate but similarly structured menu system is displayed (FIG. 3). In this display, bar 10 is "Chemistry". Bar 100 includes items Atoms, Molecules, States, Organic Compounds and Reactions. Upon selection of "Atoms", tertiary bar 200 provides select items about Particles, Valence Numbers, Nuclear Structure, Atomic Number and Isotopes. Simultaneously, the data display will present a general overview on the subject "Atoms". Again, a selection from the tertiary bar accesses for display a presentation on the topic selected. If Nuclear Structure is selected, the data display will include graphics and text on the Bohr Atom, including electron orbits, etc. Moreover, the user may retreat to high level topics, such as "States" with the system displaying general information on chemical states, while concurrently removing and replacing the tertiary items with a new group relating to this subject.

As can be appreciated, the display menus remain on screen providing lineage on the subject categories and information on the path taken to the current display. This provides a richer information environment and enhances interaction with the system. Students simply cannot resist investigating ancillary topics given the ease of access.

It should be noted that the above descriptions are presented to illustrate the invention and that modifications by those skilled in the art are possible without departing from the spirit and scope of the invention.

What is claimed is:

1. A data processing system for hierarchical access to a multilevel application comprising:

display screen controller for selective and programmed control of a two-dimensional screen display;

pointing input device for entering input events by the user, controlling a displayed cursor, and comprising a button;

a single display means for subdividing the display screen into a single simultaneous presentation composing a primary menu bar, a secondary menu bar, a tertiary menu bar, a graphics window, and a text window, wherein the primary menu bar contains a single menu item and the secondary and tertiary menu bars contain a plurality of menu items containing a label and selectable by the displayed cursor, wherein each menu bar is subdivided between the menu items, wherein primary menu bar provides a single subject menu item, wherein the secondary menu bar provides menu items corresponding to the primary categorical item with increasing level of detail, wherein the tertiary menu bar provides menu items corresponding to the secondary display bar with increasing level of detail in response to selecting a secondary menu item;

default screen display means for initially displaying blank data in graphics window, text window, and the tertiary bar until selection of a menu item;

secondary menu item display coding regimen means for displaying differing and distinct colors and patterns in the secondary menu items;

secondary bar position determination means for determining the cursor located over a secondary menu item;

tertiary menu bar accessing means for accessing tertiary menu items in response to the cursor located over a secondary menu bar item;

tertiary menu item display coding regimen means for displaying the distinct color and pattern in the tertiary menu items corresponding to the selected secondary menu item;

highlighting means for highlighting the tertiary menu item in response to a cursor located over a tertiary menu item;

activation means for activating the button on the cursor input device after highlighting a menu item;

memory for storing data files containing data for display in the graphics window and text window;

graphics window for displaying graphics, image and video data corresponding to an arbitrarily selected menu item in one of the bars and corresponding to the level of detail of the menu item;

text window for displaying text corresponding to an arbitrarily selected menu item in one of bars and corresponding to the level of detail of the selected menu item, wherein multiple pages may be scrolled; and

tertiary menu bar replacement means for replacing the tertiary menu items in response to retreating to the secondary menu bar and selecting a different secondary menu bar item.

2. The system in claim 1 including an audio output means for producing voice or music audio output.

* * * * *

US006072492A

United States Patent [19]

Schagen et al.

[11] Patent Number: 6,072,492

[45] Date of Patent: *Jun. 6, 2000

[54] **SELECTING AN INFORMATION ITEM IN AN INFORMATION PROCESSING SYSTEM**

[75] Inventors: **Cornelis M. Schagen; Suzan A. Van Delft**, both of Eindhoven, Netherlands; **Hugo M. H. Paulissen**, Hasselt, Belgium

[73] Assignee: **U.S. Philips Corporation**, New York, N.Y.

[*] Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

[21] Appl. No.: **08/968,370**

[22] Filed: **Nov. 12, 1997**

[30] **Foreign Application Priority Data**

Dec. 4, 1996 [EP] European Pat. Off. 96203431

[51] **Int. Cl.**[7] **G06F 3/14**; G06F 15/16

[52] **U.S. Cl.** **345/353**; 345/356; 345/357; 345/329; 709/217; 709/219

[58] **Field of Search** 345/353, 357, 345/356, 329, 332; 395/200.49, 200.47; 707/501; 709/219, 217

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,485,175	1/1996	Suzuki	345/353
5,581,685	12/1996	Sakurai	345/353
5,689,668	11/1997	Beaudet et al.	345/353
5,704,051	12/1997	Lane et al.	345/357
5,708,787	1/1998	Nakano et al.	345/352
5,787,417	7/1998	Hargrove	707/4
5,801,702	9/1998	Dolan et al.	345/357

OTHER PUBLICATIONS

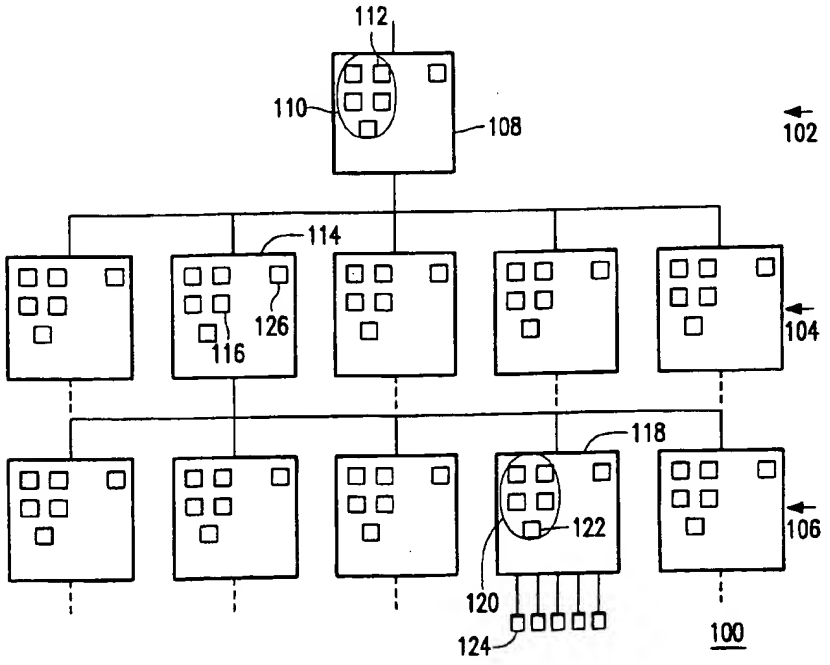
"CD–Online", Printed by CD–Online Ltd., 2 Percy St., London WIP 9FA. No Date Avail.

Primary Examiner—Raymond J. Bayerl
Assistant Examiner—Cuong T. Thai
Attorney, Agent, or Firm—Gregory L. Thorne

[57] **ABSTRACT**

An information processing system is programmed to give a user access to a first subset of information items using a first hierarchical multi-level menu-structure on a local work station and to a second such subset using a second such menu-structure on an internet server. The second menu-structure includes at least the sub-menus provided in the first menu-structure. The second information subset includes at least the information items included in the first subset. A user leaving the first menu-structure at a particular sub-menu will enter the second menu-structure in a sub-menu that corresponds to the particular sub-menu. When the user selects to access an information item in the second subset using the second menu, if a current version of the information item is available in the first subset, then unless otherwise specified, the information item in the first subset is accessed as a substitute.

19 Claims, 2 Drawing Sheets



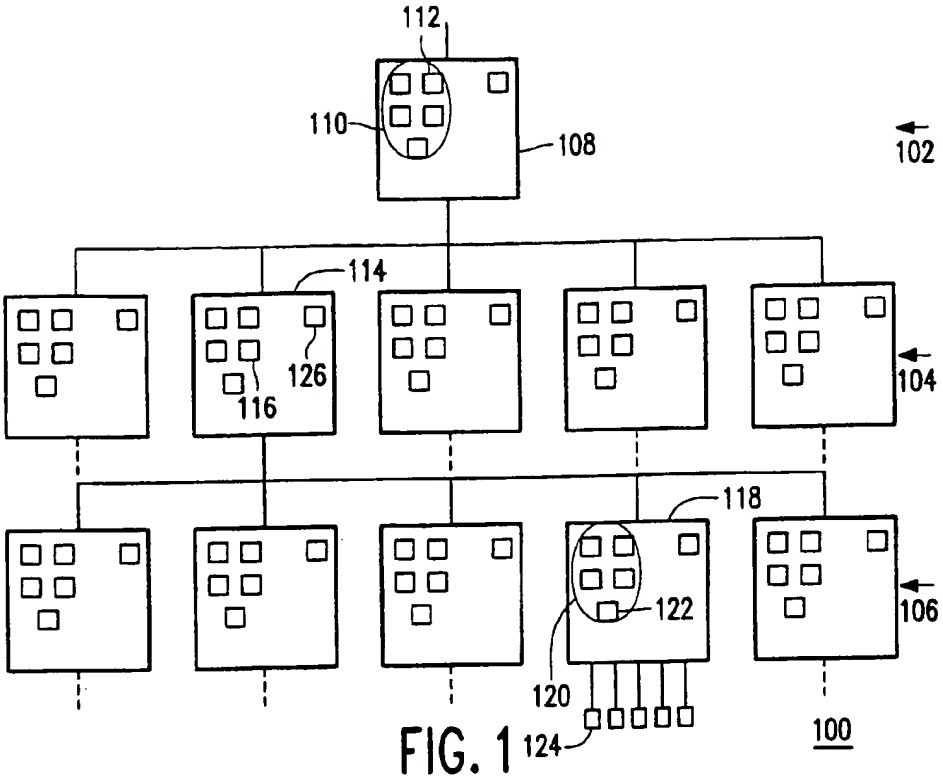

112
110
108
102
114
126
116
104
118
122
106
120
124
100

FIG. 1

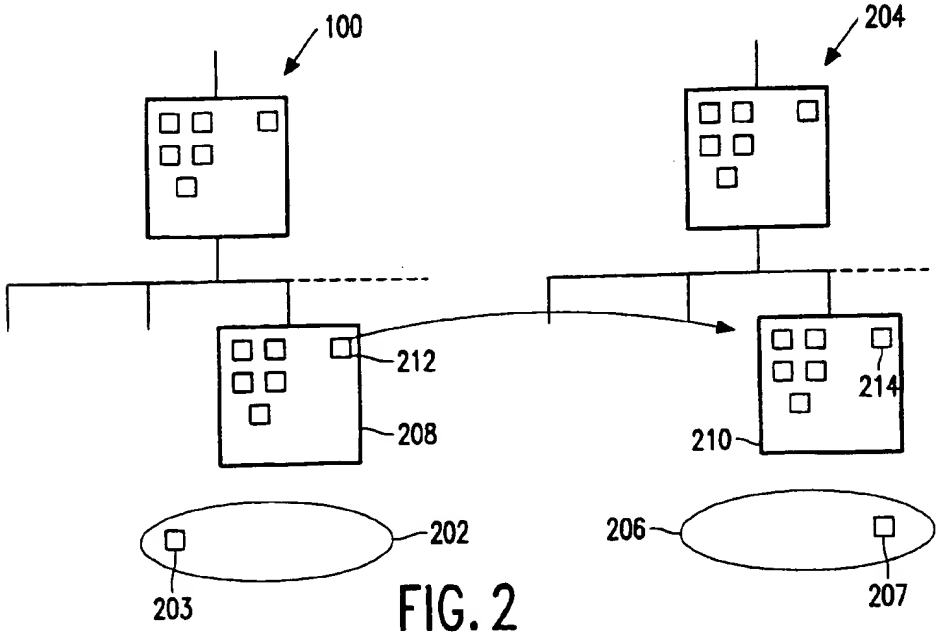

100
204
212
208
214
210
202
203
206
207

FIG. 2

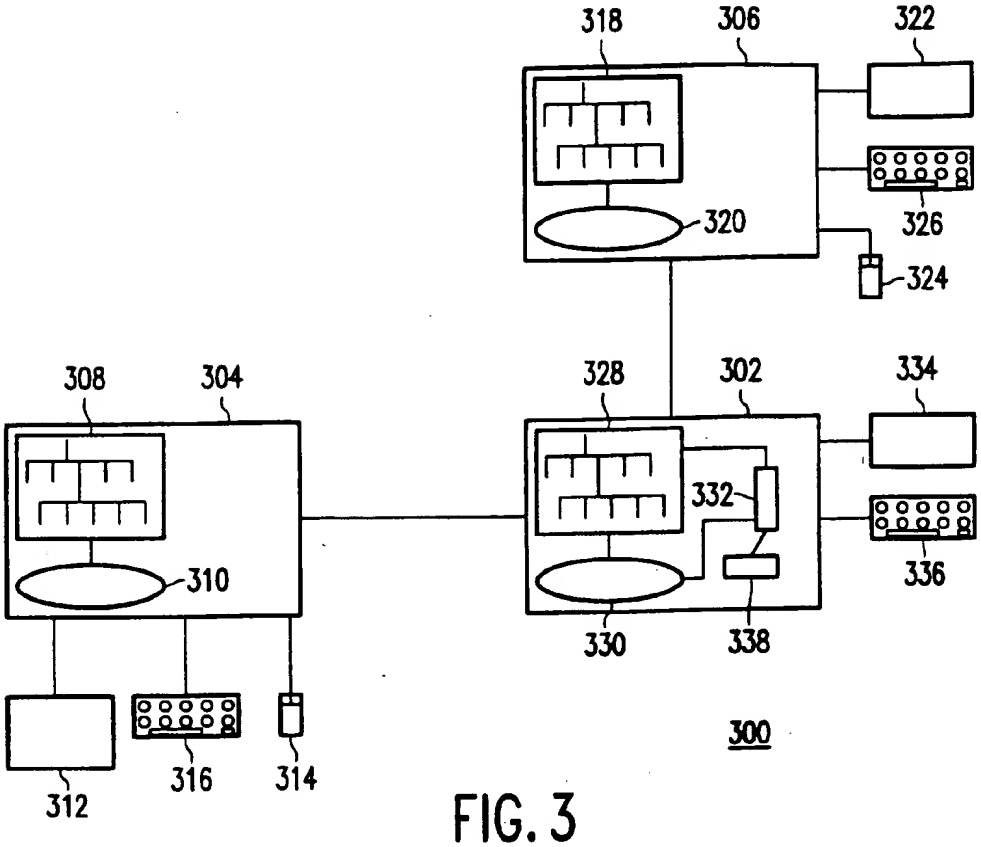

318
306
322
320
326
324
308
304
328
302
334
332
310
336
330
338
300
312
316
314

FIG. 3

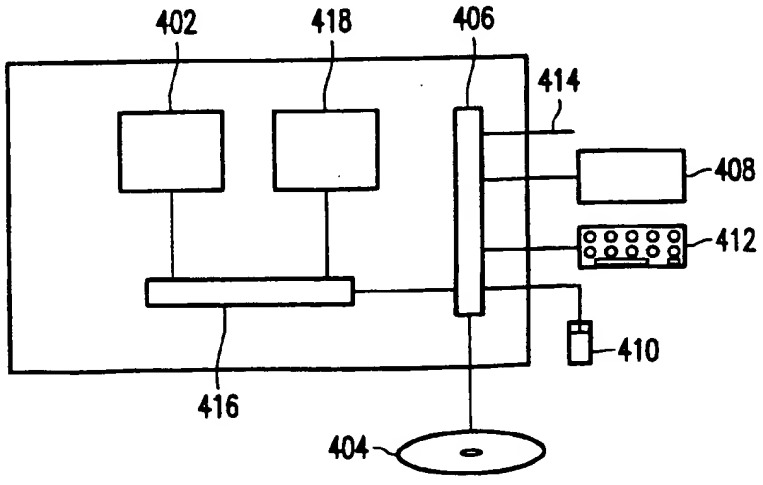

402
418
406
414
408
412
410
416
404
400

FIG. 4

SELECTING AN INFORMATION ITEM IN AN INFORMATION PROCESSING SYSTEM

FIELD OF THE INVENTION

The invention relates to the field of user interfaces in information handling systems.

Method for selecting an information item in an information processing system, system for such a method, local station in such a system, and carrier with program for such a method.

The invention relates to a method for selecting a particular information item from a plurality of information items in an information processing system, which method enables a user to select the particular information item from a first subset of the information items through a first hierarchical multi-level menu-structure and to alternatively select the particular information item from a second subset of the information items.

The invention further relates to a system for enabling a user to select a particular information item from a plurality of information items, which system comprises a first hierarchical multi-level menu-structure for selecting the particular information item from a first subset of the information items and secondary means for selecting the particular information item from a second subset of the information items.

The invention further relates to a local station for enabling a user to select a particular information item from a plurality of information items, which local station comprises a local hierarchical multi-level menu-structure for selecting the particular information item from a local subset of the information items and secondary means for selecting the particular information item from an external subset of the information items.

The invention further relates to a carrier with an executable program to be loaded into such a local station.

It is known to retrieve an information item from a plurality of information items through a hierarchical menu-structure in an information processing system. An information item contains data on a certain subject in one of various ways. For instance, an information item may contain a still image, a textual message, an audio fragment or a video film. Such a method can be applied for various kinds of subjects, e.g. for data on products, data on tourist attractions, for books and the like. In the known method, the user is offered the possibility to retrieve the desired data not only from the subset of information items accessible through the hierarchical menu-structure but also from an additional, second subset of information items. The known method is for instance used in so-called home-shopping applications, where the user has a local station with a local database with product data and where in addition the user is given access to a central station with complementary data like current price and availability of products. The CD-Online system, as described in the brochure 'CD-Online, The Net on your set' provides for a program and an infrastructure for such applications. A drawback of the known method is that the user has to navigate in two sets of information items, each requiring knowledge about the structure and accessing mechanism of the particular subset.

SUMMARY OF THE INVENTION

It is an object of the invention to provide a method of the kind set forth which offers the user an improved accessibility to the information items compared with the known method.

This object is achieved according to the invention in a method that is characterised in that the method further enables the user to proceed from a particular sub-menu in the first menu-structure to a corresponding sub-menu in a second hierarchical multi-level menu-structure for selecting the particular information item from the second subset through the second menu-structure, the second menu-structure at least comprising sub-menus corresponding to the sub-menus of the first menu-structure and the second subset at least comprising the information items of the first subset. The method according to the invention offers the user access to the second subset of information items in a way that is highly similar to the way of access to the first subset of information items. This relieves the cognitive load of the user when navigating between the first and second subset of information items, because initially the user only has to familiarise himself with a single menu-structure that underlies both the first and the second menu-structure. The second menu-structure comprises sub-menus corresponding to the sub-menus of the first menu-structure and therefore offers the user the possibility to leave the first menu-structure and to browse through the second subset of information items, with potentially updated data compared with the first subset, without the risk of getting lost in the second menu-structure. A user who leaves the first menu-structure from a particular sub-menu, will enter the second menu-structure in a sub-menu that corresponds to the particular sub-menu in the first menu-structure, which further helps the user in navigating in the two menu-structures.

An embodiment of the method according to the invention is characterised in that the method further enables the user to return from a particular sub-menu in the second menu-structure to the corresponding sub-menu in the first menu-structure. This offers the user the possibility to return to the first menu-structure, after having consulted the second menu-structure and possibly the second subset of information items. When returning from the second menu-structure to the first menu-structure, it is advantageous to return to that sub-menu in the first menu-structure which corresponds to the sub-menu of the second menu-structure when leaving that menu-structure. This further helps the user in navigating between the two subsets of information items.

An embodiment of the method according to the invention is characterised in that the first menu-structure resides on a local station and that the second menu-structure resides on a central station, to which the local station is connected. The method according to the invention can be applied in a system comprising a local station, with the first menu-structure giving the user access to the first subset of the information items, and a central station, with the second menu-structure giving the user access to the second subset of the information items. The advantage of the application of the method in such a system is that the central station can be employed by many users on different local stations, thereby giving access to the second menu-structure and the second subset of information items to these many users.

An embodiment of the method according to the invention is characterised in that the local station is connected to the central station via Internet. Since Internet is widely available and used by many people, it is advantageous to use Internet as the connection mechanism between the local station and the central station in the method according to the invention.

An embodiment of the method according to the invention is characterised in that upon selecting the particular item from the second subset, it is verified whether the first subset comprises an information item corresponding to the particular information item in the second subset and if such

information item is comprised in the first subset then this information item is accessed to substitute for the particular information item. By verifying whether the first subset of the information items comprises an information item corresponding to the particular information, i.e. an information item containing data on the same subject as the particular information item that has been selected by the user from the second subset of information items, a transfer of the data of that information item from the second menu-structure to the first menu-structure can be avoided. The data on that subject can in that case be retrieved from the information item in the first subset. This then avoids the need of sending potentially a lot of data over the network, in the case the method according to the invention is applied for a local station with the first menu-structure and a central station with the second menu-structure. Furthermore, the feature of this embodiment can be used to provide the data of the information items in the first subset in a format that is specifically adapted to the local station, whereas the corresponding data in the second subset is in a general purpose format. When a user wants to retrieve data from an information item in the second subset and the local station determines that a corresponding information item concerning the same subject is locally available in the first subset, then the local station supplies the data from the information item in the first subset instead of the data from the information item in the second subset.

An embodiment of the method according to the invention is characterised in that the second menu-structure allows modification to include a further sub-menu and/or to include an updated information item. The fact that the second menu-structure and/or the second subset of information items can be updated makes it in an easy way possible to supply the user with more recent data. This is because the first menu-structure and the first subset of the information items may remain static, i.e. the structure and items are not updated, while the user gets access to the more recent data through the second menu-structure. This feature is particularly useful, when a number of different users with their individual local stations can be connected to the same central station comprising the second menu-structure and the second subset of information items. Updating the data on this central station will then give all these users access to the more recent data. This updating includes substituting the data referred to by an information item of the second subset and includes the addition or modification of a sub-menu of the second menu-structure.

An embodiment of the method according to the invention is characterised in that the method comprises an initialisation step in which start-up information external to the first menu-structure is retrieved and that the sub-menu according to the start-up information is presented to the user as the first sub-menu. The retrieval of the start-up information, comprising an indication as to what sub-menu has to be presented first to the user, makes it possible that the user enters the first menu-structure in a at that moment most appropriate sub-menu. Furthermore, this mechanism makes it possible to present a different opening sub-menu at different times, thus creating a dynamic, adapting behaviour of the first menu-structure, while it may remain static in its internal composition.

It is a further object of the invention to provide a system of the kind set forth which offers the user an improved accessibility to the information items compared with the known system. This object is achieved according to the invention in a system that is characterised in that the secondary means comprise a second hierarchical multi-level menu-structure for selecting the particular information item from the second subset, the second menu-structure at least comprising sub-menus corresponding to the sub-menus of the first menu-structure and the second subset at least comprising the information items of the first subset, and that the system is arranged for enabling the user to proceed from a particular sub-menu in the first menu-structure to the corresponding sub-menu in the second hierarchical multi-level menu-structure for selecting the particular information item through the second menu-structure. A user in this system is supported in navigating in the two subsets of information items, because the system offers access to the second subset through a second menu-structure that is based on the same structure as the first menu-structure. Further support is offered by the system through the fact that a user, when going from the first menu-structure to the second menu-structure, will enter the second menu-structure in the sub-menu that corresponds to the sub-menu in the first menu-structure in which the user was at the moment of transfer.

An embodiment of the system according to the invention is characterised in that the system comprises a local station containing the first menu-structure and the first subset of the information items, and a central station containing the second menu-structure and the second subset of the information items, the local station being connectable to the central station. In such a system, the central station may be set up to service a plurality of local stations. The central station can be arranged to provide this service for a number of local stations simultaneously. In this embodiment, the second subset of the information items residing on the central station can be accessed by a plurality of different users.

An embodiment of the system according to the invention is characterised in that the system comprises a modification module for modifying the second menu-structure to include a further sub-menu and/or to include an updated information item. Through updating the data in the information items on the central station, all users that connect via their local station to the central station can access the updated data. This constitutes an easy mechanism in the system for providing the users in such a system with more recent data. It is not necessary to update the first subset of information items on all local stations in such a system, but it is sufficient to only update the second subset of information items on the central station.

It is a further object of the invention to provide a local station of the kind set forth which offers the user an improved accessibility to the information items compared with the known local station. This object is achieved according to the invention in a local station that is characterised in that the local station is arranged for enabling the user to proceed from a particular sub-menu in the local menu-structure to a corresponding sub-menu in an external hierarchical multi-level menu-structure for selecting the particular information item from the external subset of the information items through the external menu-structure. The local station according to the invention has the advantage that the user can leave the local menu-structure to enter an external menu-structure in a sub-menu that corresponds to the sub-menu of local menu-structure in which the user was active at the moment of leaving. This eases the navigation of the user in the external menu-structure and reduces the risk of getting lost in that menu-structure.

An embodiment of the local station according to the invention is characterised in that the local station comprises a verification module for, upon selecting the particular item

from the external subset by the user, verifying whether the local subset comprises an information item corresponding to the particular information item in the external subset and for, if such information item is comprised in the local subset, accessing this information item to substitute for the particular information item. If the user selects an information item from the external subset and if the verification shows that the local subset contains a corresponding information item, then the data desired by the user can be supplied directly from the local subset and need not to be transferred from the external subset. The avoidance of a potentially long transfer time in such a case is an advantage of the local system according to the invention. A further advantage is that the data in the local subset can be stored in a format that is particularly suitable for the local station, whereas the data stored in the external subset must be stored in a generally applicable format since potentially different types of local station must be able to receive and interpret that data. An example is that the data concerns a still image which is stored in high resolution in the local subset and in a low resolution in the second subset, because the high resolution image would take too much time to be transferred. Another example is that the data constitutes in the local subset a video track in MPEG format and in the external subset a slide show with a number of still images, mimicking a video track, because not all kinds of local stations that are serviced from the external subset can handle MPEG.

An embodiment of the local station according to the invention is characterised in that the local station is arranged to receive start-up information from an external source and to perform an initialisation step for presenting a first sub-menu to the user on the basis of the start-up information so received. Through this mechanism, the local station can offer the user on entrance a first sub-menu that is most appropriate at that time. This can for instance be used to indicate updated data in the external subset of information items. Furthermore, this mechanism can be used to offer the user another opening sub-menu from time to time without modifying the local menu-structure. This is particularly attractive in a situation where the local station is not or not frequently updated. This mechanism for providing another opening sub-menu at different times then offers a dynamic behaviour and avoids that a user gets annoyed by the same opening sub-menu time after time.

It is a further object of the invention to provide a carrier of the kind set forth containing a program which, when loaded into a local apparatus, offers the user an improved accessibility to the information items compared with the known local station. This object is achieved according to the invention in a carrier comprising an executable program to be loaded in a further general purpose apparatus for preparing the apparatus for the execution of the tasks of the local station as described above. Such a carrier could optionally comprise the local subset of the information items, although this subset could be provided to the apparatus in another way.

The above citations are hereby incorporated in whole by reference.

Those skilled in the art will understand the invention and additional objects and advantages of the invention by studying the description of preferred embodiments below with reference to the following drawings which illustrate the features of the appended claims:

BRIEF DESCRIPTION OF THE DRAWINGS

Further embodiments of the invention are recited in the dependent claims.

The invention and its attendant advantages will be further elucidated with the aid of exemplary embodiments and the accompanying schematic drawings, whereby:

FIG. 1 schematically shows an example of hierarchical menu-structure according to the invention,

FIG. 2 schematically shows a transfer from the first to the second menu-structure,

FIG. 3 schematically shows a system according to the invention, and

FIG. 4 schematically shows a local station according to the invention.

Corresponding features in the various Figures are denoted by the same reference symbols.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. 1 schematically shows an example of hierarchical menu-structure according to the invention. This menu-structure **100** has a highest level **102**, a level **104** one step down, and a lowest level **106** one step further down. Each level has a number of sub-menus giving access to the sub-menus at the next level down. Level **102** has a single sub-menu **108** which gives access to the sub-menus of level **104**. To this end, sub-menu **108** has options **110** and selection by a user of a particular option causes the with the particular option corresponding sub-menu at level **104** to be presented to the user. When, for example, a user selects option **112**, sub-menu **114** will subsequently be presented to the users. The number of options in a certain sub-menu is equal to the number of sub-menus at its next level down. In the example, sub-menu **108** has five option and level **104** has five corresponding sub-menus. Each of the sub-menus at level **104** has, in the same way as described above, a number of sub-menus that corresponds to its number of options at the next level down. In the example, sub-menu **114** has five sub-menu at level **106**. Since this is true for each of the sub-menus at level **104**, level **106** has in total **25** distinct sub-menus. At the lowest level, the options of a sub-menu do not give access to a sub-menu at a next level down but give access to a particular piece of data. So the options of a sub-menu at the lowest represent the information items that can be accessed via the menu-structure. In the example, a sub-menu at level **106** has five options, so in total the menu-structure provides access to **125** information items. These **125** information items form the set information items from which a selection can be made through the menu-structure **100**. The steps for selecting a particular information item are for instance the following. A user selects option **112** of sub-menu **108** and, as a result of that, sub-menu **114** is presented. The user then selects option **116** of that sub-menu **114** and the corresponding sub-menu **118** is presented to the user. Sub-menu **118** offers the user a choice out of the group **120** of five information items and the user then selects the particular information item **122**. After this selection the data **124** can be retrieved.

The example given comprises a hierarchical menu-structure with three levels but application of the invention is not restricted to this particular number of three and an other number of levels is possible. Furthermore, each of the sub-menus of the example comprises five options from which the user can select one particular option. This is not a restriction for the invention and an other number of options in a sub-menu may be used. It is also possible that different sub-menus comprise different numbers of options. A sub-menu with options can be implemented as a display screen with buttons, each having a label with associated text to

identify the button. The user then operates a mouse or trackball to point to the desired button and by clicking it. For instance by moving a cursor to the desired button and by subsequently pushing a mouse-button. However the sub-menu can be implemented in a different way, e.g. by presenting a screen with a numbered list with the available options and by letting the user enter the number of the desired options. The actual implementation of the hierarchical menu-structure is not essential to the application of the invention. In addition to the options described above, a sub-menu in the menu-structure according to the invention offers the user a choice to enter a second menu-structure. To this end, such a sub-menu comprises a connection option, for instance connection option **126** in sub-menu **114**, through which a user can leave the menu-structure **100** and enters another menu-structure.

FIG. **2** schematically shows a transfer from the first to the second menu-structure. A first hierarchical multi-level menu-structure **100**, as shown in FIG. **1**, provides access to a first subset **202** of the information items and a second multi-level menu-structure **204** provides access to a second subset **206** of the information items. The second menu-structure **204** is based on the first menu-structure in that it comprises at least sub-menus that correspond to the respective sub-menus of the first menu-structure. The second menu-structure may comprise additional levels in the hierarchy and may comprise additional options per sub-menu. The only requirement is that a sub-menu in the first menu-structure has a corresponding sub-menu in the second menu-structure. For example the second menu-structure comprises for the sub-menu **208** in the first menu-structure **100** a corresponding sub-menu **210**.

The first subset **202** includes a selection of the information items available to the user. A particular information item, like information item **203**, contains data on a specific subject and these data can be retrieved for the user after selection of the particular information item. The second subset **206** includes at least the information items that are also included in the first subset **202**. A particular information item included in the second subset **206**, like information item **207**, may contain data that are different from the data contained by the corresponding information item in the first subset. So an information item can be included in both subsets, like an element that is a member of two sets in the mathematical sense, while its data in the two subsets remains distinct. An information item is to be understood as an identification for the data that can be retrieved for a user. An example is an information item that is the Eiffel Tower in Paris, whereby its data in the first subset is a still image in first format (e.g. JPEG) and its data in the second subset is a still image in a second format (e.g. GIF). Then both subsets include the same information item Eiffel Tower, the first subset contains the JPEG image and the second subset contains the GIF image.

Now, a user can select a particular information item through the first menu-structure **100** or through the second menu-structure **204**. This can be applied in a system where the first menu-structure and the first subset reside on a local station and the second menu-structure and the second subset reside on a central station. In such an organisation, a user can access the first subset as long as the data in it are sufficient and the user need only connect to the central station if extra data from the second subset are required. These extra data may concern data in addition to the data in the first subset or may concern data that is more recent than in the first subset. The invention is not limited to such a system with a local station and a central station but can also be employed in other systems, e.g. where the first subset forms some kind of cache for the second subset. The second menu-structure embraces the first menu-structure in that the sub-menus of the first menu-structure have corresponding sub-menus in the second menu-structure. This makes it easier for the user to access the second subset because the user does not need to learn a different menu-structure, when switching from the first menu-structure to the second menu-structure. A sub-menu in the first menu-structure comprises a connect option, e.g. option **212** in sub-menu **208**, through which a connection is made with the second menu-structure **204** and which takes the user into this second menu-structure. According to the invention, a user enters the second menu-structure **204** in the sub-menu that corresponds to the sub-menu that he left in the first menu-structure. So a user selecting the connect option **212** in sub-menu **208** will enter the second menu-structure in sub-menu **210** because this corresponds to sub-menu **208**. This mechanism of entering at the corresponding sub-menu further helps the user in accessing the second subset and further reduces the risk of getting lost in the second menu-structure compared with the situation in which the user is confronted with a completely different method of access entered at an unknown point. This mechanism is realised by identifying the sub-menus in the first menu-structure and by identifying the corresponding sub-menus in the second menu-structure in the same way. In an embodiment, the sub-menu has a code that indicates at which level it is and what is its sequence number at that level and this code is used to access the second menu-structure in the desired, corresponding sub menu.

A sub-menu in the second menu-structure may comprise a return option, like option **214** in sub-menu **210**, that provides a return back to the first menu-structure. A user who selects the return option, will leave the second-menu structure and return to the first menu-structure in the corresponding sub-menu. A particular sub-menu in the second menu-structure could concern an additional sub-menu in that it does not have a corresponding sub-menu in the first menu-structure. If the user returns from that particular sub-menu, then it is determined which other sub-menu that has a corresponding sub-menu in the first menu-structure is closest to that particular menu and the user return to that corresponding sub-menu. The closest sub-menu could be determined on the basis that it is on the same level in the hierarchy and nearest in sequence at that level, although other arrangement may be made.

The first and the second menu-structure may be implemented and/or presented in mutually different ways. In an embodiment of the invention, the first menu-structure is implemented in a way dedicated to the local station on which it resides, in this case a CD-i player with a dedicated program running on it. The second menu-structure in this embodiment is implemented in HTML pages on a central station, which can be accessed from the local station via a suitable browser. The usage of HTML, which stands for Hyper Text Markup Language, is well known and widely used on the Internet. An HTML page comprises fields through which other HTML pages can be reached. Such a field is a link to another HTML page, the field having an associated address of that other HTML page, and upon selection of that field the browser will retrieve that other page. In this embodiment of the invention, when a user selects a connection option in a particular sub-menu on the local station, a connection will be made with the central station and the browser will retrieve the HTML page based on the identification of the particular sub-menu.

FIG. **3** schematically shows a system according to the invention. A system **300** comprises a central station **302**, a

local station **304** and a local station **306**. The local station **304** comprise a first menu-structure **308** giving access to a first subset **310** of information items. The local station **304** is further equipped with a display screen **312**, for displaying the menu-structure and other information, and input devices, for instance a mouse **314** and a keyboard **316**, for entering a selection by the user. The local station **304** can be implemented as a dedicated apparatus but could also be implemented on a general purpose apparatus that is programmed to perform the required tasks. It can be implemented on a standard Personal Computer, on a CD-i player, but also on a more powerful workstation. The subset **310** resides on a suitable storage medium, like a hard disk, a CD-ROM or other device, from which information can be retrieved. The local station may comprise further devices that are not shown and not relevant for the present invention. The local station **306** comprises a first menu-structure **318** for accessing a first subset **320**, a display screen **322**, a mouse **324** and/or a keyboard **326**. Also local station **306** can be implemented as a dedicated apparatus or as a general purpose apparatus specially programmed for this system. Local station **304** and local station **306** can be implemented in mutually different ways. The central station **302** comprises a second menu-structure **328** for accessing a second menu-structure **330**. This second menu-structure may be stored on a hard disk of the central station. The central station further comprises a modification module **332** for updating the second menu-structure **328** and/or the second subset **330**, a display screen **334** and an input device, like keyboard **336**. The central station **302** can be implemented on a general purpose computer programmed in a suitable manner to perform the required tasks. The system shown in FIG. **3** comprises two local station by way of example. The system could have only one local station but could also have a larger number of local stations that can be connected to the central station.

The central station optionally comprises start-up information **338**, to be retrieved by a local station. This start-up information contains an indication for the local station as to which of the sub-menus should be presented first to the user as an opening sub-menu. The start-up information **338** can be modified by the modification module **332**.

An embodiment of the system according to the invention concerns a system for providing information on a city. A program comprising the first menu-structure and the first subset is implemented on a CD-i disk, an optical disk dedicated for a CD-i player, that can be loaded into a CD-i player. The user of the player can consult the subset on the disk and retrieve data that appear of interest. Furthermore, the embodiment comprises an Internet server, with the second menu-structure implemented in HTML pages and the second subset organised as shown in FIG. **2**, acting as the central station. The user of the local station can connect to the server to access the second subset for obtaining additional or more recent data than he has locally available. The information on the CD-i disk can be periodically updated and new releases of the CD-i disks distributed. The information on the central station can be updated more frequently and can include additional data compared with the CD-i disk. Stations external to the system can also access the Internet server and consult the information present on the server.

FIG. **4** schematically shows a local station according to the invention. The local station **400** comprises a local menu-structure **402** that gives access to the local subset of information items. This menu-structure is shown in detail in FIG. **1**. The local subset resides on a carrier **404**, which can be an optical disk, magnetic disk or other suitable storage medium. The carrier **404** can also comprise a copy of the local menu-structure that can be loaded into the station at start-up. The local station has an interface **406** for connecting external devices to the station. These include a display screen **408**, a mouse **410** and/or a keyboard **412**. Furthermore, the interface **406** provides access to an external station via a connection symbolised by line **414**. The connection can be realised in various way, e.g. via the public telephony network or via a computer network like Internet. The local station further comprises a controller **416** to control the tasks of the various other components of the station. A user of the local station **400** can select an information item from the local subset through the local menu-structure **402**. In addition, the user can choose to leave the local menu-structure and to proceed in an external menu-structure in a sub-menu that corresponds to the sub-menu where the user left the first menu-structure.

The local station optionally includes a verification module **418**, that verifies whether an information item that is selected from an external subset is also present in the local subset. If this is the case, then the data of the information item are retrieved from the local subset instead of from the external subset. This has the advantage of avoiding the potentially time consuming transfer of the data via connection **414**. A further advantage is that the data of the information item present in the local subset are in a format that is specifically suitable for the local station, whereas the data present in the external subset must be in a general applicable format. A further option is that the local station **400** retrieves at the beginning of a session start-up information from an external source via connection **414**. This start-up information comprises an indication as to what sub-menu should be presented first to the user. In this way, the local source, which may be a central station as described above, can control the way the user enters the first menu-structure. This can be used to draw the user's attention to a particular information item or group of information items, but can also be used to create a dynamic behaviour of the first menu-structure. Even when the carrier is regularly updated, the user will consult the same carrier a number of times and it is advantageous that the system can offer a variation in the opening sub-menus presented to the user.

The local station can be realised as a special apparatus but can also be programmed on a general purpose apparatus. This can be a Personal Computer or a CD-i player in which a program specially developed to this end, can be loaded. The Personal Computer or the CD-i player thus programmed is able to perform the tasks of the local station as described above. In an embodiment of the invention, a CD-i player is employed as local station. The carrier in that case is an optical disk comprising program modules to be loaded into the CD-i player, including the menu-structure, the verification module and a browser for retrieving HTML pages from the Internet server. The carrier further comprises the local subset of information items and the data of those information items.

The invention has been disclosed with reference to specific preferred embodiments, to enable those skilled in the art to make and use the invention, and to describe the best mode contemplated for carrying out the invention. Those skilled in the art may modify or add to these embodiments or provide other embodiments without departing from the spirit of the invention. Thus, the scope of the invention is only limited by the following claims:

We claim:

1. A method comprising the steps of:

selecting a particular sub-menu using a first hierarchical multi-level menu-structure with entries corresponding to particular information items in a first subset of information items;

proceeding from the first particular sub-menu in the first menu-structure to a corresponding sub-menu in a second hierarchical multi-level menu-structure with entries corresponding to particular information items in a second subset of information items, including at least sub-menus corresponding to the sub-menus of the first menu-structure and the second subset including at least information items corresponding to information items of the first subset.

2. The method of claim 1, further comprising the step of returning from a particular sub-menu in the second menu-structure to the corresponding sub-menu in the first menu-structure.

3. The method of claim 1, further comprising the steps of: providing the first menu-structure residing on a local station; providing the second menu-structure residing on a central station; and connecting the local station with the central station.

4. The method of claim 3, in which connecting the local station to the central station includes connecting via an Internet.

5. The method of claim 1, further comprising the steps of: upon selecting the particular item from the second subset, determining whether the first subset contains an information item corresponding to the particular information item in the second subset; and if such information item is contained in the first subset, then accessing the corresponding information item in the first subset instead of the particular information item in the second subset.

6. The method of claim 1, further comprising the step of modifying the second menu-structure to include a further sub-menu and/or to include an updated information item.

7. The method of claim 1, further comprising initialization steps of retrieving start-up information external to the first menu-structure; and presenting the sub-menu according to the start-up information, to the user as the first sub-menu.

8. The method of claims 1, in which:

the method further comprises the step of returning from a particular sub-menu in the second menu-structure to a corresponding sub-menu in the first menu-structure;

the method further comprises of the steps of:

providing a first menu-structure residing on a local station; and providing the second menu-structure residing on a central station; and connecting the local station with the central station;

the connecting step includes connecting via the Internet;

the method further comprises the steps of: upon selecting the particular item from the second subset, determining whether the first subset contains a current version of the information item corresponding to the particular information item in the second subset; and if such corresponding information item is contained in the second subset, then accessing the corresponding information item in the first subset instead of the particular information item in the second subset;

the method further comprises the step of modifying the second menu-structure to include a further sub-menu and/or to include an updated information item;

the method further comprises initialization steps of: retrieving start-up information external to the first menu-structure; and presenting the sub-menu according to the start-up information to the user as the first sub-menu;

the second menu-structure includes at least the sub-menus of the first menu-structure; and the second subset includes at least the information items in the first subset; and

the method further comprises the step of loading programs into a more general purpose system to provide a system for this method for selecting particular information items.

9. A system comprising:

a first hierarchical multi-level menu-structure with entries corresponding to particular information items in a first subset of information items; and

a second hierarchical multi-level menu-structure with entries corresponding to particular information items in a second subset of information items; and

means for proceeding from a particular sub-menu in the first menu-structure to a corresponding sub-menu in the second menu-structure for selecting a particular information item from the second subset using the second menu-structure

wherein the second menu-structure contains at least sub-menus corresponding to the sub-menus of the first menu-structure and the second subset contains at least information items of the first subset.

10. The system of claim 9, further comprising: a local station containing the first menu-structure and the first subset of the information items; a central station containing the second menu-structure and the second subset of the information items; and means for connecting the local station to the central station.

11. The system of claim 9, further comprising means including a verification module for, upon selecting the particular item from the second subset by the user, determining whether the first subset contains an information item corresponding to the particular information item in the second subset and for, if such information item is contained in the first subset, accessing the corresponding information item in the first subset instead of the particular information item in the second subset.

12. The system of claim 9, further comprising means including a modification module for modifying the second menu-structure to include a further sub-menu and/or to include an updated information item.

13. The system of claim 9, in which:

the system further comprises: a local station containing the first menu-structure and the first subset of the information items; a central station containing the second menu-structure and the second subset of the information items; and means for connecting the local station to the central station;

the system further comprises verification means for, upon selecting the particular item from the second subset by the user, determining whether the first subset includes an information item corresponding to the particular information item in the second subset, and for, if such information item is contained in the first subset, accessing the corresponding information item in the first subset instead of the particular information item in the second subset;

the system further comprises modification means for modifying the second menu-structure to include a further sub-menu and/or to include an updated information item; the second menu-structure includes at least the sub-menus of the first menu-structure; and the second subset includes at least the information items in the first subset; and

the system further comprises means to load programs into the system for converting a general purpose system into the system for selecting the information items.

14. A local station comprising:

a local hierarchical multi-level menu-structure with entries corresponding to particular information items in a local subset of information items;

means for proceeding from a particular sub-menu in the local menu-structure to a corresponding sub-menu in an external hierarchical multi-level menu-structure with entries corresponding to particular information items in an external subset of information items; and

means for selecting a particular information item from the external subset of the information items using the external menu-structure,

wherein the external menu-structure contains at least sub-menus corresponding to the sub-menus of the local menu-structure and the external subset contains at least information items of the local subset.

15. The local station of claim **14** further comprising means including a verification module for, upon selecting the particular item from the external subset by the user, determining whether the local subset contains an information item corresponding to the particular information item in the external subset and for, if such information item is contained in the local subset, accessing the corresponding information item in the local subset instead of the particular information item in the external subset.

16. The local station of claim **14**, further comprising means to receive start-up information from an external source and to perform an initialization step for presenting a first sub-menu to the user on the basis of the start-up information so received.

17. The station of claim **14**, in which:

the local station further comprises verification means for, upon selecting a particular information item from the external subset by the user, determining whether the local subset includes a current version of the information item corresponding to the particular information item in the external subset, and for, if such current information item is contained in the local subset, accessing the corresponding information item in the local subset instead of the particular information item in the external subset;

the local station further comprises means to receive start-up information from an external source upon initialization, and means for presenting a first sub-menu to the user upon the initialization, depending on the start-up information received upon the initialization;

the external menu-structure includes at least the sub-menus of the local menu-structure; and the external subset includes at least the information items in the local subset and

the station further comprises means to load executable programs to provide programmed apparatus for converting a general purpose computer into the local station.

18. An information carrier comprising an executable program to be loaded in an apparatus for providing in the apparatus programmed structures including:

a local hierarchical multi-level menu-structure with entries corresponding to particular information items in a local subset of information items;

means for proceeding from a particular sub-menu in the local menu-structure to a corresponding sub-menu in an external hierarchical multi-level menu-structure with entries corresponding to particular information items in an external subset of information items; and

means for selecting a particular information item from the external subset of the information items using the external menu-structure,

wherein the external menu-structure contains at least sub-menus corresponding to the sub-menus of the local menu-structure and the external subset contains at least information items of the local subset.

19. Apparatus to generate signals to program a system to implement a process comprising:

selecting a particular sub-menu using a first hierarchical multi-level menu-structure with entries corresponding to particular information items in a first subset of information items;

proceeding from the first sub-menu in the first menu-structure to a corresponding sub-menu in a second hierarchical multi-level menu-structure with entries corresponding to particular information items in a second subset of information items,

wherein the second menu-structure contains at least sub-menus corresponding to sub-menus of the first menu-structure and the second subset of information items contains at least information items corresponding to all the information items of the first subset.

* * * * *

Draker

US005666477A

United States Patent [19]

Maeda

[11] Patent Number: 5,666,477

[45] Date of Patent: Sep. 9, 1997

[54] **METHOD AND APPARATUS FOR SETTING GRAPH DEFINITION ITEMS IN GRAPH PROCESSING SYSTEM**

[75] Inventor: **Jun Maeda**, Shizuoka, Japan

[73] Assignee: **Fujitsu Limited**, Kawasaki, Japan

[21] Appl. No.: **719,416**

[22] Filed: **Sep. 24, 1996**

Related U.S. Application Data

[63] Continuation of Ser. No. 320,212, Oct. 11, 1994, abandoned.

[30] **Foreign Application Priority Data**

Jan. 4, 1994 [JP] Japan 6-000051

[51] **Int. Cl.**[6] **G06T 11/20**

[52] **U.S. Cl.** **345/440**; 345/353

[58] **Field of Search** 395/140, 352-354, 395/356; 345/133, 134, 139-140, 146

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,611,306	9/1986	Crehan et al.	395/156
4,813,013	3/1989	Dunn	395/159
4,967,381	10/1990	Lane et al.	364/551.01
5,226,118	7/1993	Baker et al.	395/161
5,375,201	12/1994	Davoust	395/161
5,557,716	9/1996	Oka et al.	395/140
5,581,677	12/1996	Myers et al.	395/140
5,581,678	12/1996	Kahn	395/140

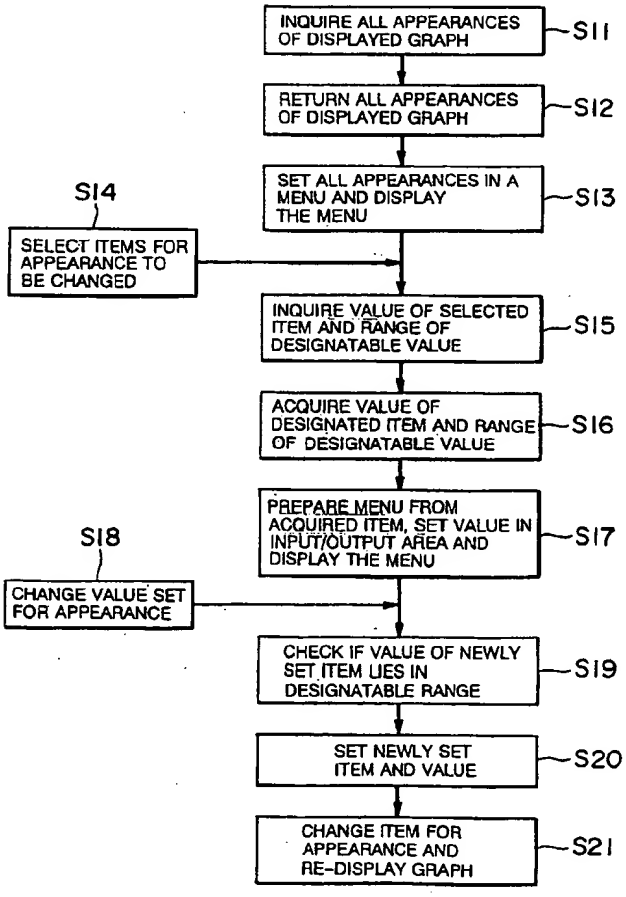
Primary Examiner—Joseph H. Feild

Attorney, Agent, or Firm—Staas & Halsey

[57] **ABSTRACT**

When setting of items to define a graph is requested through an information input section, a menu preparing section, which has an interactive editing function, inquires settable definition items of one of a plurality of graph generating sections, and prepares a menu for an interactive edition to set the settable definition items based on information on the settable definition items which is returned from the associated one of the graph generating sections.

15 Claims, 5 Drawing Sheets



02/03/2003, EAST Version: 1.03.0002

FIG.1

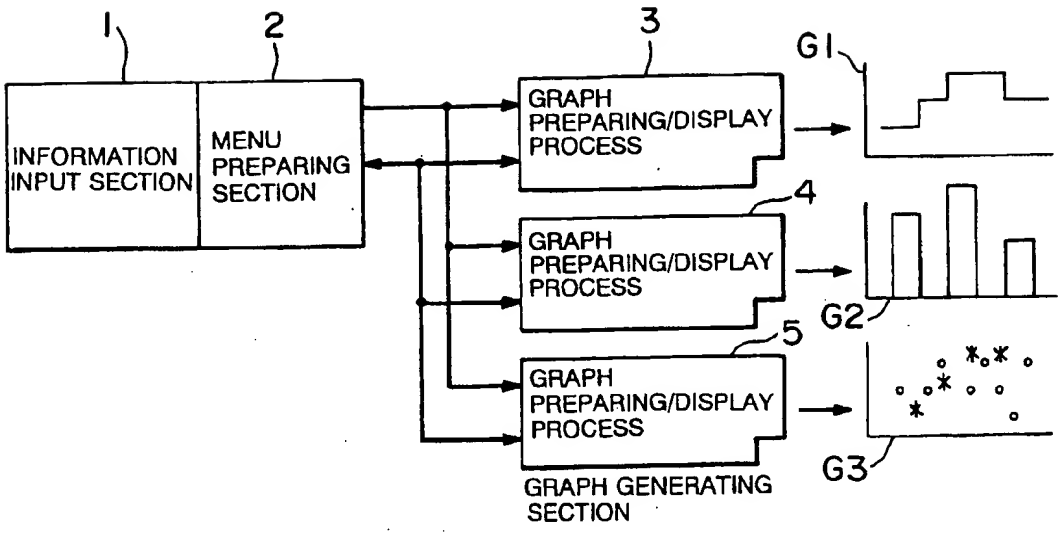

1
2
3
INFORMATION INPUT SECTION
MENU PREPARING SECTION
GRAPH PREPARING/DISPLAY PROCESS
GRAPH PREPARING/DISPLAY PROCESS
GRAPH PREPARING/DISPLAY PROCESS
GRAPH GENERATING SECTION
G1
4
G2
5
G3

FIG.2

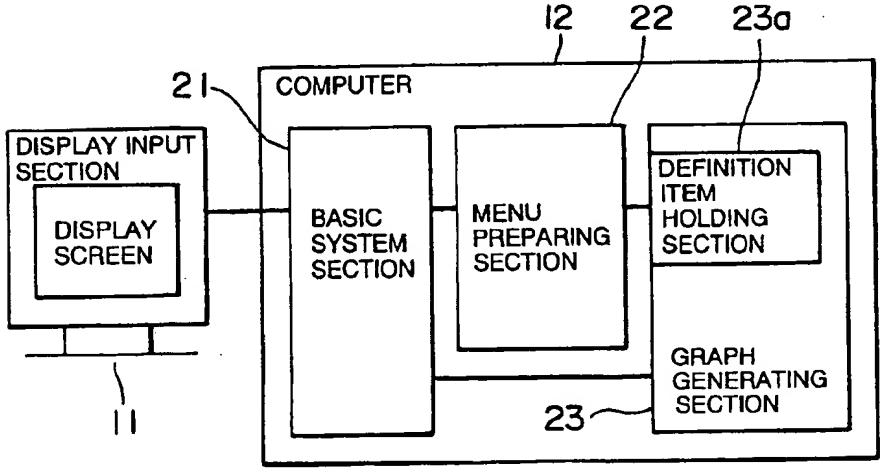

12
22
23a
21
COMPUTER
DISPLAY INPUT SECTION
DISPLAY SCREEN
BASIC SYSTEM SECTION
MENU PREPARING SECTION
DEFINITION ITEM HOLDING SECTION
GRAPH GENERATING SECTION
23
11

FIG.3

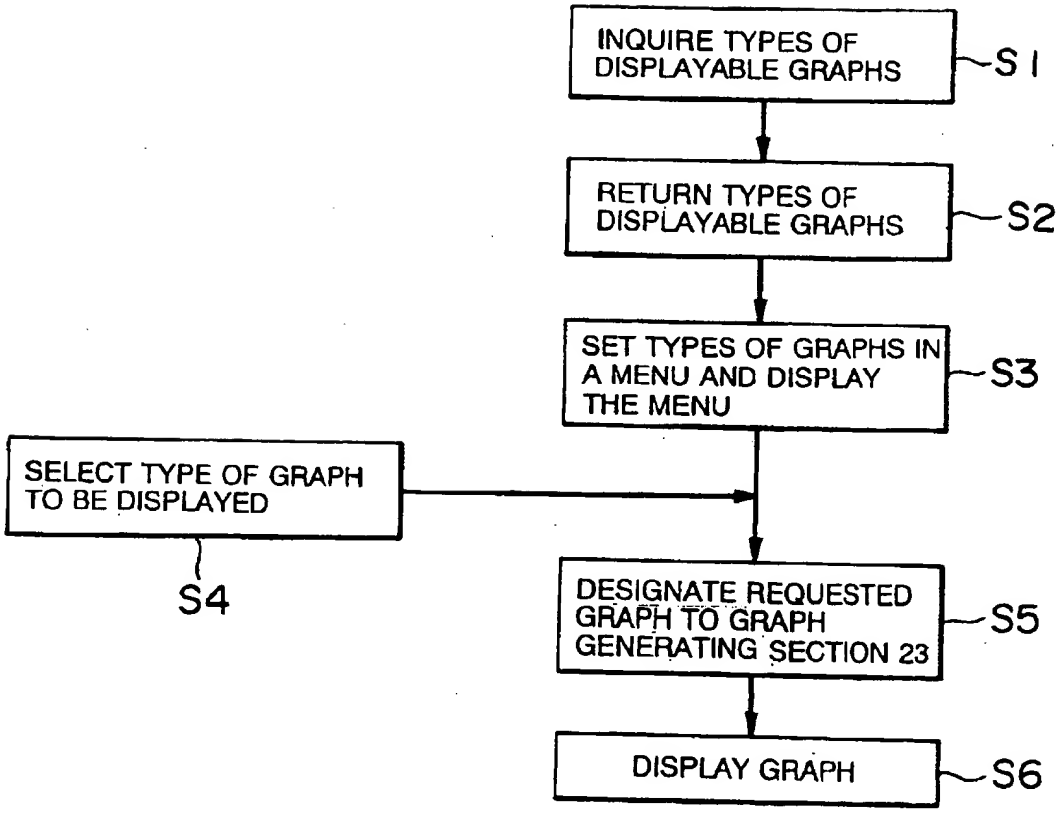

INQUIRE TYPES OF DISPLAYABLE GRAPHS
S1
RETURN TYPES OF DISPLAYABLE GRAPHS
S2
SET TYPES OF GRAPHS IN A MENU AND DISPLAY THE MENU
S3
SELECT TYPE OF GRAPH TO BE DISPLAYED
S4
DESIGNATE REQUESTED GRAPH TO GRAPH GENERATING SECTION 23
S5
DISPLAY GRAPH
S6

FIG.4

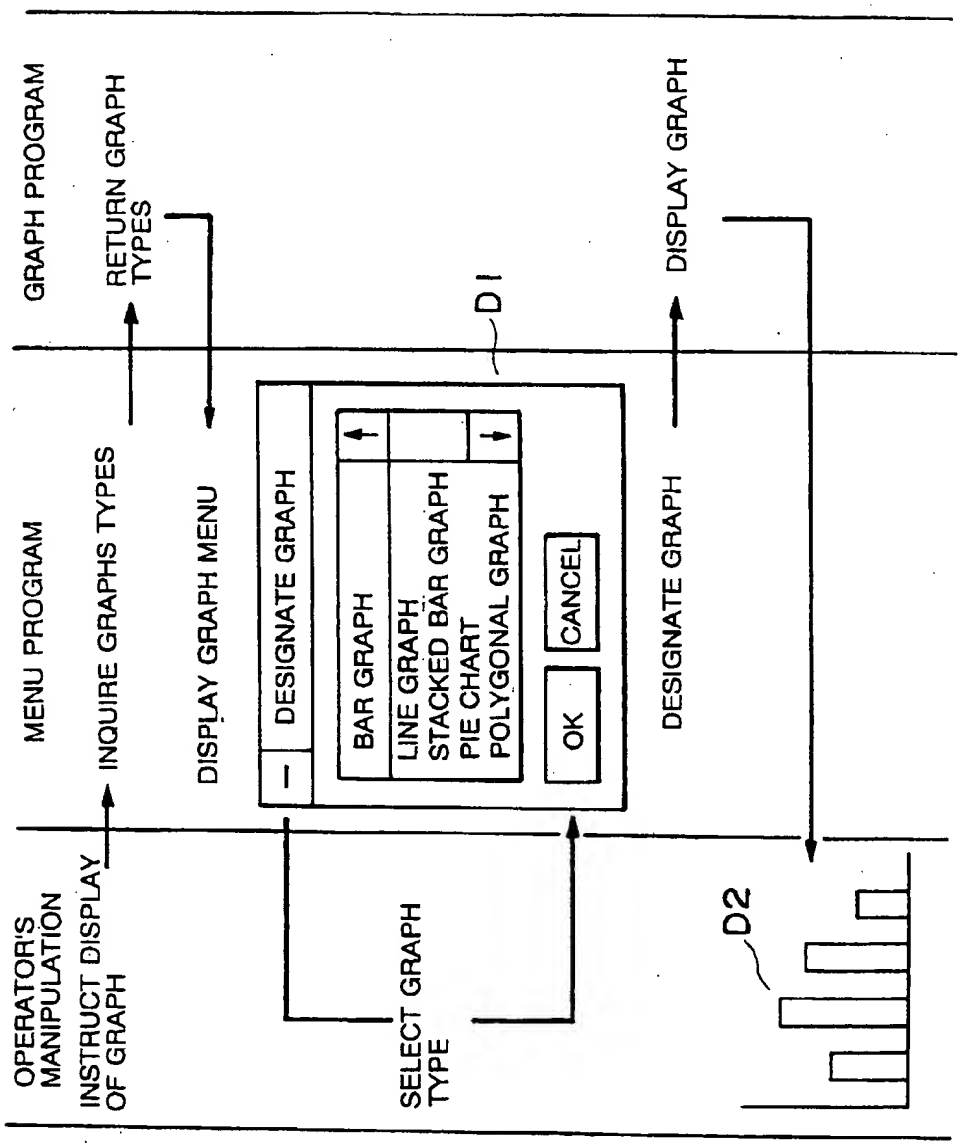

GRAPH PROGRAM
RETURN GRAPH TYPES
DISPLAY GRAPH
D1
MENU PROGRAM
INQUIRE GRAPHS TYPES
DISPLAY GRAPH MENU
DESIGNATE GRAPH
BAR GRAPH
LINE GRAPH
STACKED BAR GRAPH
PIE CHART
POLYGONAL GRAPH
OK
CANCEL
DESIGNATE GRAPH
OPERATOR'S MANIPULATION
INSTRUCT DISPLAY OF GRAPH
SELECT GRAPH TYPE
D2

FIG.5

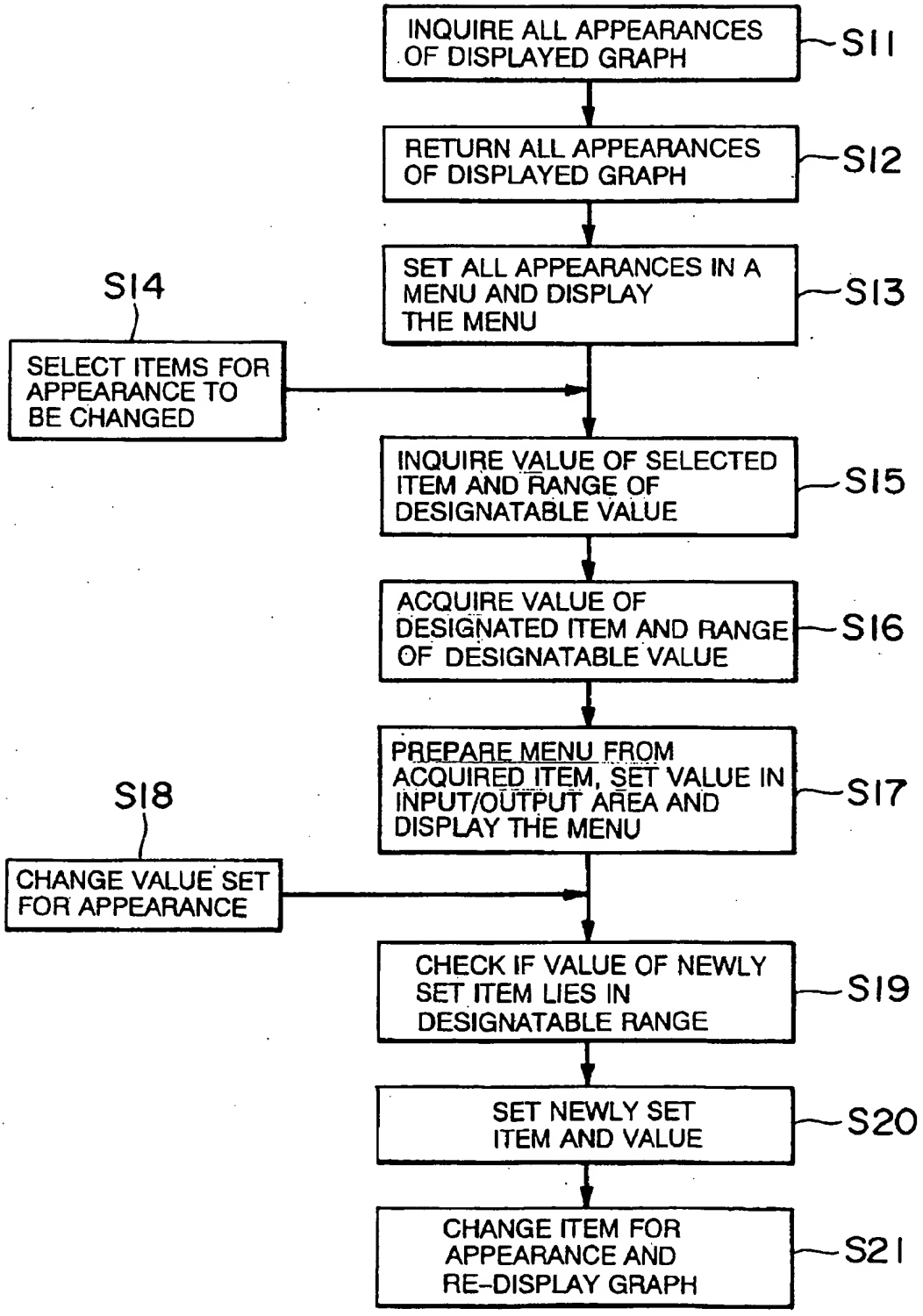
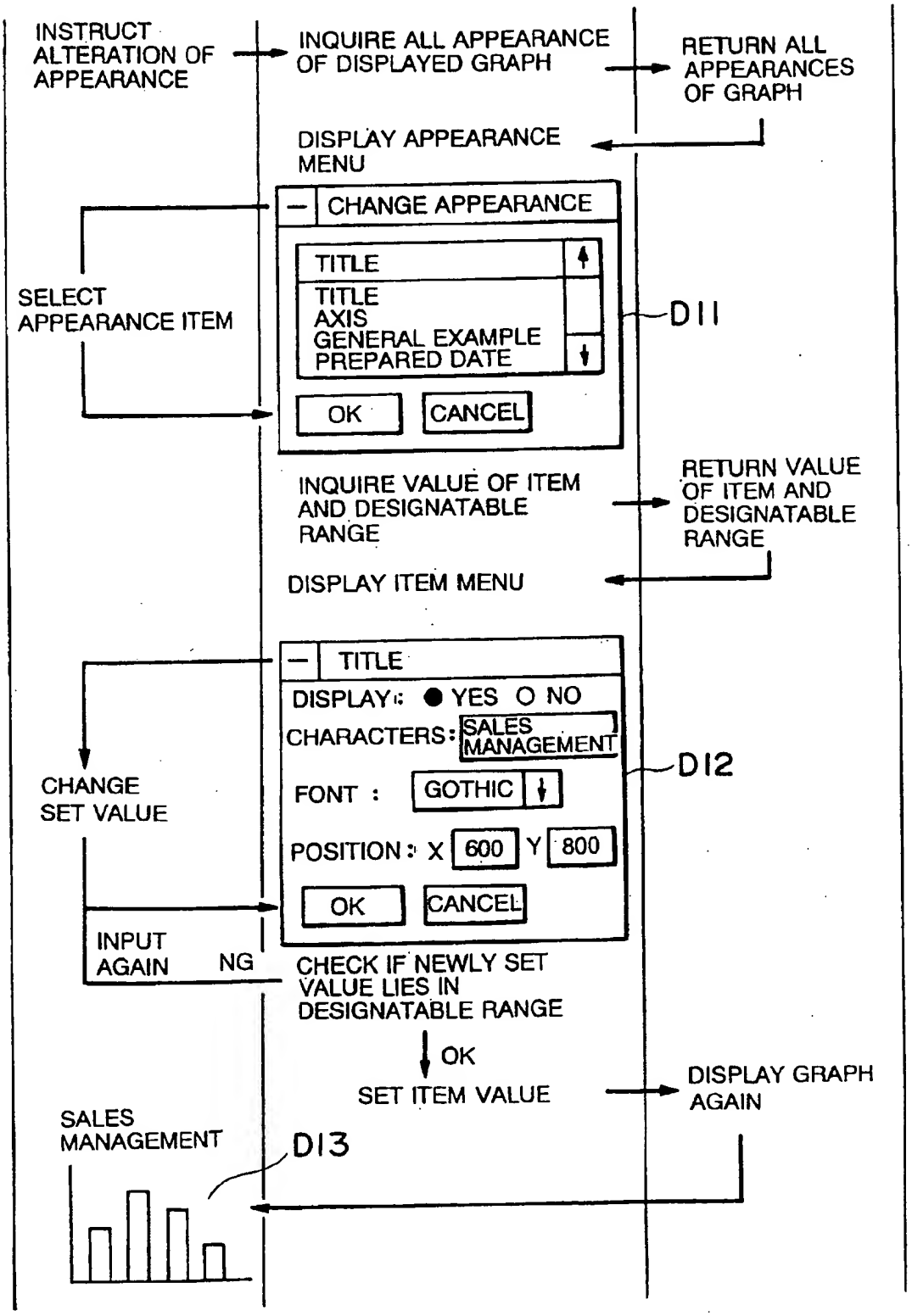

INQUIRE ALL APPEARANCES OF DISPLAYED GRAPH
S11
RETURN ALL APPEARANCES OF DISPLAYED GRAPH
S12
SET ALL APPEARANCES IN A MENU AND DISPLAY THE MENU
S13
S14
SELECT ITEMS FOR APPEARANCE TO BE CHANGED
INQUIRE VALUE OF SELECTED ITEM AND RANGE OF DESIGNATABLE VALUE
S15
ACQUIRE VALUE OF DESIGNATED ITEM AND RANGE OF DESIGNATABLE VALUE
S16
PREPARE MENU FROM ACQUIRED ITEM, SET VALUE IN INPUT/OUTPUT AREA AND DISPLAY THE MENU
S17
S18
CHANGE VALUE SET FOR APPEARANCE
CHECK IF VALUE OF NEWLY SET ITEM LIES IN DESIGNATABLE RANGE
S19
SET NEWLY SET ITEM AND VALUE
S20
CHANGE ITEM FOR APPEARANCE AND RE-DISPLAY GRAPH
S21

FIG.6


INSTRUCT ALTERATION OF APPEARANCE
INQUIRE ALL APPEARANCE OF DISPLAYED GRAPH
RETURN ALL APPEARANCES OF GRAPH
DISPLAY APPEARANCE MENU
CHANGE APPEARANCE
TITLE
TITLE
AXIS
GENERAL EXAMPLE
PREPARED DATE
OK
CANCEL
D11
SELECT APPEARANCE ITEM
INQUIRE VALUE OF ITEM AND DESIGNATABLE RANGE
RETURN VALUE OF ITEM AND DESIGNATABLE RANGE
DISPLAY ITEM MENU
TITLE
DISPLAY: YES NO
CHARACTERS: SALES MANAGEMENT
FONT : GOTHIC
POSITION: X 600 Y 800
OK
CANCEL
D12
CHANGE SET VALUE
INPUT AGAIN
NG
CHECK IF NEWLY SET VALUE LIES IN DESIGNATABLE RANGE
OK
SET ITEM VALUE
DISPLAY GRAPH AGAIN
SALES MANAGEMENT
D13

METHOD AND APPARATUS FOR SETTING GRAPH DEFINITION ITEMS IN GRAPH PROCESSING SYSTEM

This application is a continuation of application Ser. No. 08/320,212, filed Oct. 11, 1994, now abandoned.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a graph processing system which prepares and displays graphs, and, more particularly, to a method and apparatus for setting graph definition items to define the appearance of a graph in an interactive manner, i.e., for newly setting graph definition items or changing the setting of old graph definition items in an interactive manner.

2. Description of the Related Art

Graphs or charts are used to express, for example, statistical numeric data in various fields. Recently, multifarious forms of graphs are available so that a user selectively uses one or more proper forms of graphs according to the purpose.

In this respect, it is demanded that graph processing systems aided by a computer or the like should be able to not only prepare and display multifarious forms of graphs but also easily set the fine appearance of each graph. It is therefore necessary to finely set many items to define the appearance or the like of each graph.

To set items to define the appearance or the like of a graph, i.e., to execute new setting or alteration of setting, the definition items are generally set using a menu system in an interactive manner. This requires the preparation of menus which allow the fine setting many definition items for the appearances or the like of individual graphs.

In a conventional graph processing system, definition items for individual graphs are held in a graph processing section graph by graph. Therefore, a menu-based interactive editing function exists for each graph, so that to newly set or change the definition items for the appearance or the like of a graph, the interactive editing function which matches with the form of that graph should be invoked.

In executing the graph processing of three types of graphs, a step-formed graph(hereinafter referred to "step graph"), a bar graph and a scatter graph, for example, those three types of graphs are prepared and displayed in a step graph preparing/display section, a bar graph preparing/display section and a scatter graph preparing/display section respectively. Likewise, there are three menu-based interactive editing functions for setting (newly setting or altering the setting of) the definition items for those three graphs, an interactive editing function for step graphs, an interactive editing function for bar graphs and an interactive editing function for scatter graphs, for the respective types of graphs.

In this case, interactive control for allowing an operator to perform an interactive input operation is commonly executed for the three types of graphs, but the menu-based interactive editing function and the graph preparing/display process are executed for each graph.

As mentioned above, the conventional graph processing system executes the menu-based interactive editing function and the graph preparing/display process graph by graph. To add a new graph processing function or add a new definition item for any graph for which a graph processing function already exists, therefore, a new interactive editing function should be added or the processing content of the associated interactive editing function should be altered. In this case, the memory capacity inevitably increases to hold information about the added or changed processing content.

SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a method and apparatus for setting graph definition items in a graph processing system, which are able to finely set the definition items of each graph without incorporating an interactive editing function graph by graph and facilitate the addition of a new graph processing function or a new definition item.

A method and apparatus for setting graph definition items in a graph processing system according to the present invention are designed in such a manner that when setting (new setting or alteration of setting) of items to define a graph is requested, the method and apparatus inquire settable definition items of a graph generating section and prepare a menu for an interactive edition to set the settable definition items in an interactive manner based on information on the settable definition items which is returned from the graph generating section. Accordingly, it is possible to finely set the definition items of each graph without incorporating an interactive editing function graph by graph and facilitate the addition of a new graph processing function or a new definition item.

The graph definition item setting system in the graph processing system according to this invention has an information input section, a menu preparing section and first to third graph generating sections.

The information input section has an interactive control function for an operator to allow the operator to input information in an interactive manner.

The menu preparing section has an interactive editing function, inquires settable definition items of any one of the first to third graph generating sections when setting of items to define a graph is requested through the information input section, and prepares a menu for an interactive edition to set the settable definition items based on information on the settable definition items which is returned from that graph generating section.

In response to the inquiry, the graph generating section informs the menu preparing section of items constituting each graph and their currently set values. More specifically, the menu preparing section is informed of items constituting a graph, such as the types and colors of lines if the graph is a line graph or the colors of rectangular areas if the graph is a bar graph, and their currently set values.

The first, second and third graph generating sections respectively prepare and display graph data of a step graph, a bar graph and a scatter graph based on given data. Those first to third graph generating sections previously store settable definition items for the respective graphs to cope with the inquiry from the menu preparing section.

This system sets items to define a graph in the following procedures.

When setting of items to define a graph is requested through the information input section, the menu preparing section inquires settable definition items of one of the graph generating sections.

In response to this inquiry, the associated graph generating section returns the settable definition items, held therein, to the menu preparing section.

The menu preparing section prepares a menu for setting the settable definition items based on the information on the

definition items returned from the associated one of the graph generating sections.

It is to be noted that in response to the item setting request, the graph generating sections return only settable definition items to the menu preparing section.

The graph generating sections may return settable definition items in an hierarchical form so that the menu preparing section prepares a hierarchical menu according to the hierarchical definition items. In this case, if common setting elements are grouped so that an interactive editing process is commonly used for each group, the processing in the menu preparing section can be executed efficiently.

In the graph definition item setting system in the graph processing system according to this invention, as described above, the menu preparing-section for allowing an operator to input information in an interactive manner calls definition items for each graph and their currently set values from the associated one of the graph generating sections and prepares a menu based on the informed definition items.

As each graph generating section holds only the items constituting an associated graph, therefore, the individual graph generating sections need not incorporate menus and menu-by-menu processes. It is possible to finely set definition items for each graph. Further, to add the graph processing function for a new graph or add a new definition item to any existing graph, it is unnecessary to add a new interactive editing function to the menu preparing section or correct the existing interactive editing function.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will become more fully understood from the detailed description given hereinbelow and the accompanying drawings which are given by way of illustration only, and thus are not limitative of the present invention, and wherein:

FIG. **1** is a block diagram exemplarily illustrating the structure of a principle graph processing system according to a first embodiment of the present invention;

FIG. **2** is a block diagram showing the structure of a graph processing system according to a second embodiment of this invention;

FIG. **3** is a flowchart for explaining the graph display process of the graph processing system shown in FIG. **2**;

FIG. **4** is an exemplary diagram for explaining the graph display operation of the graph processing system shown in FIG. **2**;

FIG. **5** is a flowchart for explaining a process of altering the appearance of a graph in the graph processing system shown in FIG. **2**; and

FIG. **6** is an exemplary diagram for explaining an operation of altering the appearance of a graph in the graph processing system shown in FIG. **2**.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

First Embodiment

FIG. **1** illustrates the structure of a graph processing system incorporating a principle system of setting graph definition items according to a first embodiment of the present invention.

The graph processing system embodying the present invention uses a common interactive editing function and is designed in such a way that when setting of items to define a graph, i.e., new setting or alteration of setting of definition items, is requested, the graph processing system inquires settable definition items of graph generating sections, provided for respective graph forms, and prepares a menu for an interactive edition to set the settable definition items in an interactive manner based on information on the settable definition items which is returned from one of the graph generating sections.

The graph processing system shown in FIG. **1** has an information input section **1**, a menu preparing section **2** and first to third graph generating sections **3** to **5**.

The information input section **1** has an interactive control function for an operator to allow the operator to input information in an interactive manner.

The first, second and third graph generating sections **3** to **5** respectively prepare and display graph data of a step graph G1, a bar graph G2 and a scatter graph G3 based on given data.

The menu preparing section **2** has an interactive editing function. When setting of items to define a graph (new setting or alteration of setting) is requested through the information input section **1**, the menu preparing section **2** inquires settable definition items of one of the first to third graph generating sections **3–5** which associated with the designated graph form, and prepares a menu for an interactive edition to set the settable definition items based on information on the settable definition items which is returned from the first, second or third graph generating section **3**, **4** or **5**.

In response to the inquiry, the graph generating section **3**, **4** or **5** informs the menu preparing section **2** of definition items constituting each graph and their currently set values. More specifically, the menu preparing section **2** is informed of items constituting a graph, e.g., the types and colors of lines if the graph is a line graph or the colors of rectangular areas if the graph is a bar graph, as the definition items, and their currently set values.

The first to third graph generating sections **3** to **5** previously store settable definition items for the respective graphs G1 to G3 to cope with the inquiry from the menu preparing section **2**.

The graph processing system in FIG. **1** sets items to define a graph in the following steps.

(1) When setting (new setting or alteration of setting) of items to define a graph is requested through the information input section **1**, the menu preparing section **2** inquires settable definition items of one of the graph generating sections **3–5**.

(2) In response to this inquiry, the associated one of the graph generating sections **3–5** returns the settable definition items, held therein, to the menu preparing section **2**.

(3) The menu preparing section **3** prepares a menu for setting the settable definition items based on the information on the definition items returned from the associated graph generating section **3**, **4** or **5**.

It is to be noted that in response to the item setting request, the graph generating sections **3–5** return only settable definition items and their currently set values to the menu preparing section **2**.

The graph generating sections **3–5** may return settable definition items in an hierarchical form so that the menu preparing section **2** prepares a hierarchical menu according to the hierarchical definition items. In this case, if common setting elements are grouped so that an interactive editing process is commonly used for each group, the processing in the menu preparing section **2** can be executed more efficiently.

In the graph processing system in FIG. 1, the menu preparing section 2 for allowing an operator to input information in an interactive manner calls definition items for each graph and their currently set values from the associated one of the graph generating sections 3–5 and prepares a menu based on the obtained definition items.

As each of the graph generating sections 3–5 holds only the items constituting an associated graph, therefore, the individual graph generating sections 3–5 need not incorporate menus and menu-by-menu processes.

As apparent from the above description, it is possible to finely set definition items for each graph and it is unnecessary to add a new interactive editing function to the menu preparing section 2 or correct the existing interactive editing function in order to add the graph processing function for a new graph or add a new definition item to any existing graph.

Second Embodiment

FIG. 2 shows the structure of a graph processing system incorporating a system of setting graph definition items according to a second embodiment of this invention.

The graph processing system shown in FIG. 2 is the graph processing system in FIG. 1 which is designed more specifically.

The graph processing system shown in FIG. 2 comprises a display input section 11 and a computer 12.

The display input section 11 has a display, a keyboard and a pointing device all for an interactive operation to input information.

The computer 12 executes a graph preparing process. To execute this graph preparation, the computer 12 has a basic system section 21, a menu preparing section 22 and a graph generating section 23.

The basic system section 21 executes basic programs, such as an operating system and a windows system. The interactive control for allowing the operator to input information via the display input section 11 in an interactive manner is executed by this basic system section 21.

The graph generating section 23, which is the functional portion of a graph generating program, prepares and displays plural types of graph data based on given data. The graph generating section 23 has a definition item holding section 23*a* in which settable definition items for defining the appearances or the like of graphs are held graph by graph. The definition items of each graph are arranged in a hierarchical structure in accordance with the contents of the definitions, and are grouped in such a way that the items are commonly handled for common setting elements.

Data for preparing graph data that is given to this graph generating section 23 is extracted from, for example, a data base or is prepared by a so-called spreadsheet program.

The menu preparing section 22 is the functional portion of a menu preparing program. When setting of definition items for a graph is requested through the interactive operation by the operator, the menu preparing section 22 inquires settable definition items of the graph generating section 23, and prepare a menu for an interactive edition to set the settable definition items based on information on the settable definition items which is returned from the graph generating section 23. This menu preparing section prepares a hierarchical menu for the hierarchical definition items and prepares the same menu for the grouped and common setting elements.

The preparation/display of graphs and the alteration of the appearances of graphs in the graph processing system in FIG. 2 will now be discussed specifically.

Preparation/Display of Graphs

An operation for preparing/displaying graphs will now be described with reference to FIGS. 3 and 4.

FIG. 3 illustrates a flowchart for the process which is associated with the graph preparation/display process, and FIG. 4 presents an exemplary diagram for the operation which is associated with the graph preparation/display process.

In FIG. 4, the operation of the display input section 11 by the operator is shown on the left-hand side, the process of the menu preparing section 22 by the menu program is shown on the center, and the process of the graph generating section 23 by the graph program is shown on the right-hand side to illustrate their mutual relation.

The preparation and display of graphs are executed as follows.

(1) Through the operator's manipulation, the menu preparing section 22 receives a graph display request from the display input section 11 via the basic system section 21.

(2) The menu program of the menu preparing section 22 inquires the types of displayable graphs of the graph generating section 23 (step S1). The graph program of the graph generating section 23 returns information on the types of the displayable graphs to the menu preparing section 22 (step S2). In this case, the graph generating section 23 holds the information on the types of displayable graphs in the definition item holding section 23*a*, and the graph program of the graph generating section 23 obtains the information on the types of displayable graphs from the definition item holding section 23*a* and returns the information to the menu preparing section 22.

(3) The menu program of the menu preparing section 22 sets the types of displayable graphs in a menu based on the information from the graph program, and displays the menu as shown in a display screen D1 in FIG. 4 (step S3).

(4) The operator operates the display input section 11 to select the type of a graph to be displayed (step S4). The selected information is given via the basic system section 21 to the menu preparing section 22.

(5) The menu program of the menu preparing section 22 designates the type of the graph, selected and requested by the operator, e.g., a bar graph, to the graph program of the graph generating section 23 (step S5).

(6) The graph program of the graph generating section 23 displays the graph of the requested type, a bar graph D2 in FIG. 4 (step S6).

Alteration of Appearance

An operation for changing the appearance of a graph will now be described with reference to FIGS. 5 and 6.

FIG. 5 shows a flowchart for the process which is associated with a process of altering the appearance of a graph, and FIG. 6 presents an exemplary diagram for the process which is associated with an operation of altering the appearance of a graph.

In FIG. 6, as in FIG. 4, the operation of the display input section 11 by the operator is shown on the left-hand side, the process of the menu preparing section 22 by the menu program is shown on the center, and the process of the graph generating section 23 by the graph program is shown on the right-hand side to illustrate their mutual relation.

The alteration of the appearance of a graph is executed as follows.

(1) Through the operator's manipulation, the menu preparing section 22 receives an appearance changing request from the display input section 11 via the basic system section 21.

(2) The menu program of the menu preparing section **22** inquires the definition items of all the appearances of the currently selected and displayed graph of the graph generating section **23** (step S11).

In this case, the graph generating section **23** holds the information on the definition items of all the appearances of this graph in the definition item holding section **23***a*, and the graph program of the graph generating section **23** obtains the information on the definition items of all the appearances of this graph from the definition item holding section **23***a* and returns this information to the menu preparing section **22** (step S12).

(3) The menu program of the menu preparing section **22** prepares a menu for displaying all the definition items obtained from the graph generating section **23** and selectively setting the definition items as needed, and displays the menu as shown in a display screen D11 in FIG. **6** (step S13).

(4) The operator operates the display input section **11** to select the definition items to be changed (step S14). The selected information is given via the basic system section **21** to the menu preparing section **22**.

(5) The menu program of the menu preparing section **22** inquires the values of the selected definition items and the range of designatable values of the graph program of the graph generating section **23** (step St5). In this case, suppose that "title" is selected as a definition item. The graph program of the graph generating section **23** obtains the value of the selected definition item or "title" and the information of the range of the designatable value from the definition item holding section **23***a*, and returns the obtained information to the menu preparing section **22** (step S16).

(6) The menu program of the menu preparing section **22** prepares an item menu from the acquired information on the definition item, sets a value in the input/output area in the menu and displays the resultant menu as indicated by a display screen D12 in FIG. **6** (step S17).

(7) The operator manipulates the display input section **11** to change the value set for the appearance according to the menu (step S18). For example, the title is changed to "SALES MANAGEMENTS."

(8) The menu program of the menu preparing section **22** checks if the value of the newly set definition item falls within the designatable range (step S19), and gives the value of the newly set definition item to the graph program of the graph generating section **23** if this value lies in the designatable range (step S20).

If the value of the newly set definition item is out of the designatable range, the menu program informs the operator of an error and requests the operator to input a new value again.

(9) The graph program of the graph generating section **23** displays a bar graph D13, titled "SALES MANAGEMENT" as in FIG. **6**, on the display input section **11** via the basic system section **21**.

As apparent from the above, the definition items constituting each graph should only be held in the graph generating section **23** and the menu preparing section **22** has only to obtain the definition items from the graph generating section **23** to prepare the necessary menu. It is therefore unnecessary to incorporate menus and menu-by-menu processes in the menu preparing section **22**. Accordingly, the process of altering the set information does not require many resources.

Even if the process for preparing/displaying a graph is extended, it is necessary to add only the definition items to be held in the definition item holding section **23***a* of the graph generating section **23** and the process of the interactive editing function of the menu preparing section **22** need not be changed at all.

In the graph processing system in FIG. **2**, as described above, when setting of definition items to define a graph is requested, this system inquires settable definition items of the graph generating section **23** and prepares a menu for an interactive edition to set the settable definition items based on the information on the settable definition items which is returned from the graph generating section **23**. Accordingly, it is possible to finely set the definition items of each graph without incorporating an interactive editing function graph by graph and facilitate the addition of a new graph processing function or a new definition item.

The definition items include the filling, the type of hatching, the plane color, the frame color, the type of the reference line, the color of the reference line, the width of the reference line, the presence/absence of a coupling line, the type of the coupling line, the color of the coupling line, and the width of the coupling line for a bar graph, and include the type of the line, the color of the line, the width of the line, the presence/absence of a marker, the type of the marker, the color of the marker, the filling, the type of hatching, the plane color, and the frame color for a line graph. As settable items differ graph by graph, the items that are returned to the menu preparing section **22** from the graph generating section **23** may be limited to those settable items for each graph, thus improving the efficiency of the process.

If there are many definition items, it is difficult to display all the items at a time and prepare a menu for setting the definition items. In this case, the definition items should be arranged in a hierarchical structure as mentioned earlier and the definition items for the next higher level should be acquired and be displayed in a menu after each selection. Since the selection of colors, the selection of line types, the selection of hatching, etc. are common to many definition items, the process can be simplified if the menu preparing section **22** uses a common display for the menu items, thus improving the operability of the menu selection.

The sharing of the display of menu items in the menu preparing section **22** may also be carried out for definition items corresponding to similar setting elements as well as for common definition items to make common processing as much as possible.

What is claimed is:

1. A method of setting graph definition items to define a graph in a graph processing system having a graph generating section for preparing and displaying graph data based on given data and an information input section for allowing an operator to input information to said graph generating section in an interactive manner, said method comprising the steps of:

requesting the settable graph definition items from the graph generating section by a common menu preparation section used by the graph generating section for generating a menu when setting of the settable graph definition items to define a graph is requested by an inquiry through the information input section;

returning the settable graph definition items from the graph generating section in response to the inquiry; and

preparing a menu for setting the settable graph definition items by the common menu preparation section using the settable graph definition items returned from the graph generating section and common menu information.

2. The method according to claim 1, wherein said returning step returns only the settable graph definition items in response to the definition item setting inquiry.

3. The method according to claim 1, wherein said returning step returns the settable graph definition items in a hierarchical form, and said menu preparing step prepares a hierarchical menu in accordance with the settable graph definition items in hierarchical form.

4. The method according to claim 3, wherein the settable graph definition items in hierarchical form are grouped for common setting elements in said returning step and said menu preparing step.

5. An apparatus for setting graph definition items in a graph processing system, comprising:

information input means for allowing an operator to input information in an interactive manner;

a plurality of graph generating means each for storing the settable graph definition items and preparing and displaying graph data based on given data; and

common menu preparing means for requesting the settable graph definition items of at least one of said graph generating means when setting of the settable graph definition items to define a graph is requested through said information input means, and preparing a menu for interactive editing to set the settable graph definition items using the settable graph definition items requested from said graph generating means and common menu preparation information stored in said common menu preparing means.

6. The apparatus according to claim 5, wherein each of said graph generating means includes means for returning only the settable graph definition items in response to a request for the settable graph definition items from said common menu preparing means.

7. The apparatus according to claim 5, wherein each of said graph generating means includes means for returning the settable graph definition items in a hierarchical form in response to a request for the settable graph definition items from said common menu preparing means, and

wherein said common menu preparing means includes means for preparing a hierarchical menu in accordance with the settable graph definition items in hierarchical form.

8. The apparatus according to claim 7, wherein said common menu preparing means includes means for preparing a common menu for common setting elements.

9. An apparatus for setting graph definition items in a graph processing system, comprising:

a display input section for providing a display output and allowing an operator to perform an input operation;

a graph generating section for storing the settable graph definition items, preparing graph data for each type of plural types of graphs based on given data, and displaying said graph data on said display input section;

a common menu preparing section to request the settable graph definition items from said graph generating section when setting of the settable graph definition items to define a graph is requested through said information input section, preparing a menu for interactive editing to set the settable graph definition items using the settable graph definition items requested from said graph generating section and common menu preparation information stored in said common menu preparing section, and to display the menu on said display input section.

10. The apparatus according to claim 9, wherein said graph generating section returns only the settable graph definition items in response to a request for the settable graph definition items from said common menu preparing section.

11. The apparatus according to claim 9, wherein said graph generating section includes means for returning the settable graph definition items in a hierarchical form in response to a request for the settable graph definition items from said common menu preparing means, and said common menu preparing means includes means for preparing a hierarchical menu in accordance with the settable graph definition items in hierarchical form.

12. The apparatus according to claim 11, wherein said common menu preparing means includes means for preparing a common menu for common setting elements.

13. A method for processing graph definition items in a computer, comprising the steps of:

(a) storing common menu preparation information used only by a common menu preparation module;

(b) storing settable graph definition items in a graph generating module;

(c) receiving the settable graph definition items by the common menu preparation module from the graph generating module; and,

(d) generating a graph editing menu by the common menu preparation module using only the settable definition items received in step (c) and the common menu preparation information stored in step (a).

14. A graph processing system in a computer, comprising:

a memory to store common menu preparation data;

a plurality of graph generating units, each having modifiable graph definition items; and

a common menu preparation unit using the modifiable graph definition items and the common menu preparation data to generate a graph editing menu when each of said plurality of graph generating units edits the modifiable graph definition items.

15. A graph processing system, comprising:

a memory to store common menu preparation data; and

a processor to execute a plurality of graph generating modules and a common menu preparation module, each of the plurality of graph generating modules having modifiable graph definition items, and the common menu preparation module using the modifiable graph definition items and the common menu preparation data stored in said memory to generate a graph editing menu when each of said plurality of graph generating modules edits the modifiable graph definition items.

* * * * *